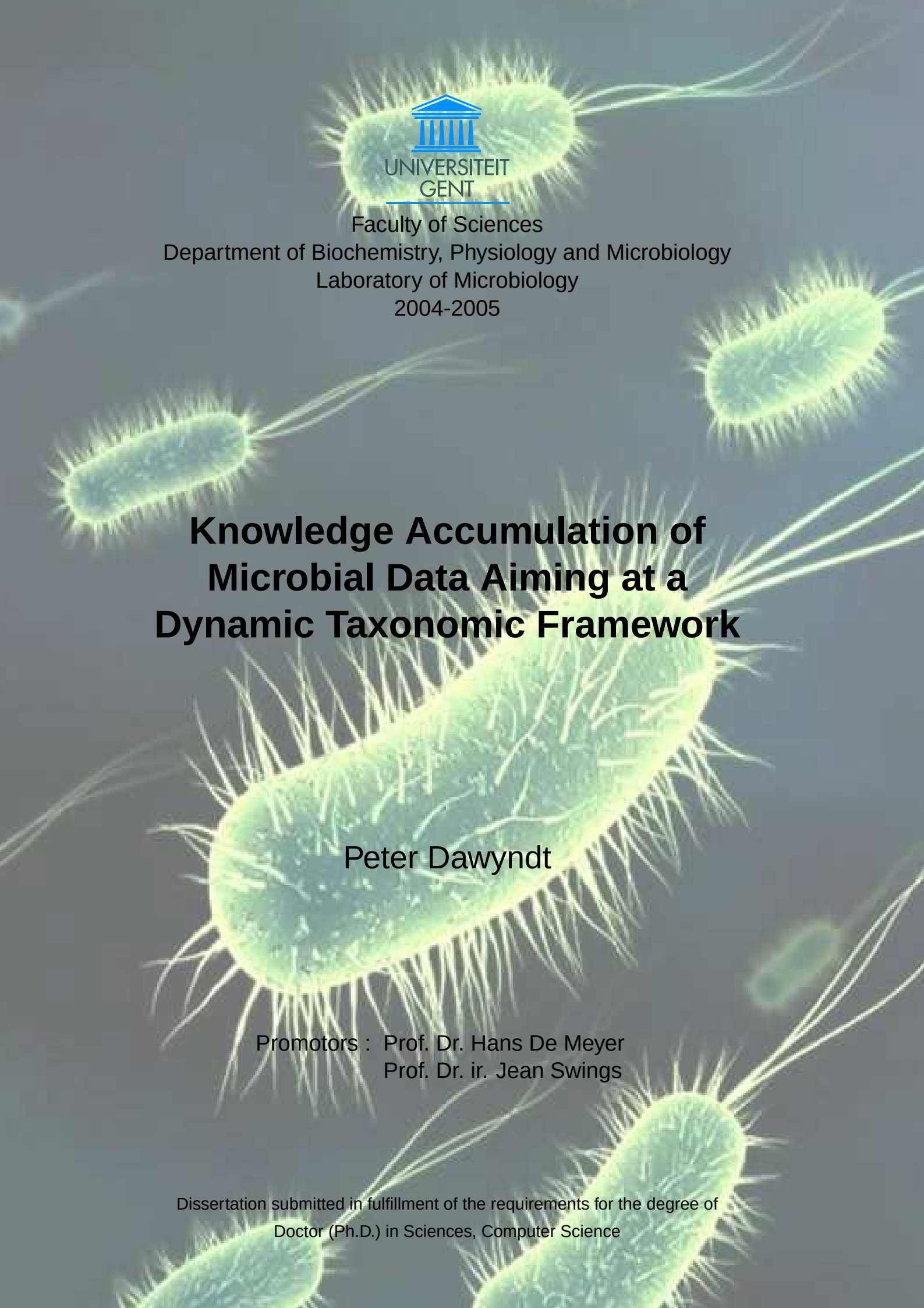


UNIVERSITEIT GENT

Faculty of Sciences
Department of Biochemistry, Physiology and Microbiology
Laboratory of Microbiology
2004-2005

Knowledge Accumulation of Microbial Data Aiming at a Dynamic Taxonomic Framework

Peter Dawyndt

Promotors : Prof. Dr. Hans De Meyer
Prof. Dr. ir. Jean Swings

Dissertation submitted in fulfillment of the requirements for the degree of Doctor (Ph.D.) in Sciences, Computer Science

UNIVERSITEIT GENT

Faculty of Sciences
Department of Biochemistry, Physiology and Microbiology
Laboratory of Microbiology
2004-2005

Knowledge Accumulation of Microbial Data Aiming at a Dynamic Taxonomic Framework

Peter Dawyndt

Promotors : Prof. Dr. Hans De Meyer
Prof. Dr. ir. Jean Swings

Dissertation submitted in fulfillment of the requirements for the degree of Doctor (Ph.D.) in Sciences, Computer Science

A Journey Through Life

And I think over again
My small adventures
When with a shore wind
I drifted out in my kayak
And thought I was in danger.

My fears,
Those small ones
That I thought so big
for all the vital things
I had to get and to reach.

And yet, there is only
One great thing,
The only thing:
To live to see in huts and on journeys
The great day that dawns
And the light that fills the world.

Song of the Kitlinguharmiut (Copper Eskimo), from the report of the Fifth Thule Expedition (1921-1924)

Members of the reading committee

Prof. Dr. Brian Austin
School of Life Sciences, Heriott-Watt University, Edinburgh, Scotland

Prof. Dr. Bernard De Baets
Department of Applied Mathematics, Biometrics and Process Control,
Ghent University, Belgium

Prof. Dr. Hans De Meyer (promotor)
Department of Applied Mathematics and Computer Science, Ghent University, Belgium

Prof. Dr. Mats Gyllenberg
Department of Mathematics and Statistics, University of Helsinki, Finland

Prof. Dr. Timo Koski
Department of Mathematics, Linköping Institute of Technology,
Linköping University, Sweden

Prof. Dr. ir. Jean Swings (promotor)
BCCM™/LMG Bacteria Collection & Laboratory of Microbiology,
Department of Biochemistry, Physiology and Microbiology,
Ghent University, Belgium

Other members of the examination committee

Prof. Walter Bossaert
Department of Applied Mathematics and Computer Science, Ghent University, Belgium

Prof. Dr. Armand De Clercq
Department of Applied Mathematics and Computer Science, Ghent University, Belgium

Prof. Dr. Albert Hoogewijs (chairman)
Department of Pure Mathematics and Computer Algebra, Ghent University, Belgium

Prof. Dr. Micah Krichevsky
United Stated Federation of Culture Collections (USFCC)
Bionomics International, Wheaton, MD, USA

Dr. Marc Vancanneyt
BCCM™/LMG Bacteria Collection,
Department of Biochemistry, Physiology and Microbiology, Ghent University, Belgium

Dr. Luc Vauterin
Applied Maths BVBA, Sint-Martens-Latem, Belgium

Contents

List of Figures

List of Tables

List of Abbreviations

ABCD	Access to Biological Collection Data
AFLP	Amplified Fragment Length Polymorphism
ANN	Artificial Neural Network
ARDRA	Amplified rDNA Restriction Analysis
BRC	Biological Resource Centre
CABRI	Common Access to Biotechnological Resources and Information
DNA	Deoxyribonucleic Acid
DDBJ	DNA Data Bank of Japan
ECL	Equivalent Chain Length
EMBL	European Molecular Biology Laboratory
FAFLP	Fluorescent Amplified Fragment Length Polymorphism
FAME	Fatty Acid Methyl Ester
GBIF	Global Biodiversity Information Facility
GC	Gas Chromatographic
GLA	Generalized Lloyd Algorithm
HMO	Hypothetical Median Organism
INSD	International Nucleotide Sequence Database
KDD	Knowledge Discovery in Databases
LIMS	Laboratory Information Management System
MALDI-TOF-MS	Matrix-Assisted Laser Desorption Ionisation Time-Of-Flight Mass Spectrometry
MDS	Multi-Dimensional Scaling
MGR	Microbial Genetic Resource
MINE	Microbial Information Network Europe
MIS	Microbial Identification System
MLST	Multilocus Sequence Typing
MLSA	Multilocus Sequence Analysis
MSC	Minimization of Stochastic Complexity
ODBC	Open DataBase Connectivity
OLAP	Online Analytical Processing
OLTP	Online Transactional Processing
PCA	Principal Component Analysis
PCR	Polymerase Chain Reaction
PFGE	Pulsed-Field Gel Electrophoresis

RAPD	Randomly Amplified Polymorphic DNA
RFLP	Restriction Fragment Length Polymorphism
RNA	Ribonucleic Acid
SC	Stochastic Complexity
SDS-PAGE	Sodium Dodecyl Sulphate Polyacrylamide Gel Electrophoresis
SI	Similarity Index
SIMCA	Soft Independent Modeling of Class Analogy
SOM	Self-Organizing Map
SQL	Structured Query Language
SVM	Support Vector Machine
TDQM	Total Data Quality Management
TSBA	Trypticase Soy Broth Agar
UPGMA	Unweighted Pair-Group Method using Arithmetic Averages

Acknowledgments

”At last the three companions turned away, and never again looking back they rode slowly homewards; and they spoke no word to one another until they came back to the Shire, but each had great comfort in his friends on the long grey road.”

— J. R. R. Tolkien

WHEN asked to make an analogy for the scientific endeavour that I have undertaken as a mathematician and computer scientist, to gradually work myself into the wondrous world of microorganisms, I would definitely pick the famous spinning plates trick. You’ve probably also admired this dance with gravity as a kid, just as I did, somewhere on television or live in a circus theatre. As the music begins to play, the performer runs onstage carrying a stack of china plates and stands in front of a long set of head-high metal rods. One at a time, the performer places a plate on a rod and sets it spinning – shaking the rod to establish momentum – and then moves on to the next rod. At some point in this process, the plate that was first in line starts to tip and looks as if it will fall. The performer leaps over to the rod under it and shakes it again to continue the plate’s progress. As more and more plates start to spin, the complexity of keeping all of the plates on the rods increases, but the plates never fall. The performer finally completes the line of plates after many near-disasters and the camera cuts to a wide-shot of the entire assembly, spinning perfectly. Wild applause ensues from the audience.

The main difference, however, is that while the plate trick only takes one performer, a much greater number of people has lend a helping hand to keep the momentum going during my scientific research. Many of the insights and ideas I have gathered about bacterial taxonomy, mathematics and computer science could never have matured without the constant flow of questions, suggestions, and constructive criticism of many friends and colleagues. Some worked as an inspiration and stimulation to push back my scientific frontiers, some contributed ideas for technical topics, and some just made life more fun while I was (not) working. It is mainly due to the credit and skills of these people that this doctoral dissertation has reached the end without any broken china. Thanks are therefore due.

Special thanks go to Jean Swings, whose visionary ideas and extensive circle of scientific acquaintances have brought this interdisciplinary project into existence, and kept it alive througout all the agonies and thriumps with which the roads of working on a PhD seem to be paved. Jean, it was not always straightforward to figure out what action should be undertaken as a response to your sometimes cryptic suggestions – yes you often remined me of the Oracle of Delphi – but at least the open-mindedness of our many discussions have been a constant challenge, and strongly determined the course of my work.

Also many thanks to Hans De Meyer, who really was there at the cradle of my scientific career. I still remember as if it were yesterday how you asked about my plans for the future during the reception that was held after I got my licentiate degree in computer science. At that time, all that was in the pipeline were the preparations for traveling to yet another distant part of the world. So, you introduced me to your friend Jean who was investigating microbes, and was searching for some collaboration with people from the mathematics field. I think it was my complete ignorance on microorganisms, rather than anything else, that made me curious about this proposition. Two years later our roads crossed again, when we started collaborating on the hierarchical clustering algorithms, together with Bernard De Baets. Hans and Bernard, without your help and suggestions, this book would have been harder to understand, contained more errors, been slightly less complete, and probably been a little shorter. And I would have missed all the fun we had in Varenna.

The exciting thing about interdisciplinary science is that it may take quite some effort for the specialists of a given research domain to find a common vocabulary for explaining the interpretation of domain-dependent problems and solutions to the non-specialists. Much of this I have learned the hard way during several meetings with the BioMaths workgroup, so I also would like to cordially thank Brian Austin, Mats Gyllenberg and Timo Koski for sharing their knowledge and opinions about practically anything in between Bayesian classification and Scottish whisky. It took us quite some time to put our collaborative efforts on the right tracks, but it is good to see that we've finally accomplished quite some amazing things, and I strongly believe we will benefit even more from it somewhere in the near future. Our forthcoming open workshop in Ghent might be the first sign of that. I am grateful to Tatu Lund for teaching me the ins and outs of the BinClass software package.

Some of the best ideas to be found in these pages were prompted by the penetrating questions and inspiring comments from the colleagues and co-workers back home in the Laboratory of Microbiology, although most of them have never stopped wondering what on earth a mathematician was doing in their laboratory. In the first place a lot of my appreciation goes to Marc Vancanneyt, whose interest and experience were simply invaluable for building up the centralized data repository we have been working on during the last couple of years. Marc, you were always available for some of-the-record discussion or explanation, notwithstanding the fact that you are probably one of the hardest-working people in the laboratory. Tackling the centralisation and cleanup of the fatty acid database has been a good lesson for getting to know some of the practice of microbiology, and I would like to thank Cindy Snauwaert for punctually transferring the data into the central database. I would also like to address a big vote of thanks to Joris Mergaert, who has been sharing the same office with me during my first years in the laboratory, and told me

so many interesting stories about the bacteria and the people who are investigating them. Joris, you are missed, both as an amazing scientist and as a good friend. Geert and Renata Huys-Coopman, thanks for the many discussions during lunch (sorry Renata but we love the sport of it), for keeping me up-to-date with the latest evolutions in the music scene, for inviting us on your agrarian homeground, for the barbecues, and for all the little things. Peter 'Dammy' Vandamme, you took reponsability for introducing me to the people in the laboratory and for explaining how proteins gels are treated, but foremost I would like to thank you for constantly reminding me that work and pleasure can go hand in hand and for helping to keep the '*Thank God it's Friday*'-sessions alive and kicking. Fabiano Thompson, thank you for the fruitful collaboration and the nice discussions. Expect me in Brasil anytime, and remember to carry out your promise of giving me a few surfing lessons, so that we can catch some tropical waves together. Margo Cnockaert and Klaas D'Haene, thanks for cheering up the coffee breaks, and for being the right hand of Santa Claus. Paul Segers, thank you for being around during the quite hours in the laboratory and for taking life as it comes. Geert Kindt, many thanks for all your efforts when computers were not doing what they are supposed to do, which seems to happen more than often. In addition to the people mentioned above, I would also like to thank Griet Casteleyn, Tom Coenye, Jeaninne De Jaeger, Fernand Depoorter, Roberto Gelsomino, Dirk Dewettinck, Elke De Clerck, Paul De Vos, Dirk Gevers, Johan Goris, Ben Lanoot, Liesbeth Masco, Sabri Naser, Virginie Storms, Pavel Svec, Kim Heylen, Jeroen Heyrman, Bart Hoste, Danny Janssens, Karel Kersters, Urbain Torck, Liesbeth Lebbe, Claudine Vereecke, Moniek Gillis, Miet Martens, Anne Willems, Antoine Benoot – and not to forget the ones I forgot right now – for helping me out with all my questions and day-to-day worries. And of course, I finally would like to thank Annemie Struyvelt, who took care of things exactly as we knew she would – masterfully and without a hint of complaint. Many of us wouldn't have enjoyed work as much as we do without your helping hand. Annemie, you're the best !

I must admit that before I started studying mathematics, I pretty much expected to meet the average profile of the super-intelligent, unwordly Einstein-brains, as these people are usually perceived within the outside world. Luckily, I soon enough became to find out that there were enough outliers in the field, and I wouldn't have expected to meet so many exciting personalities that I can still consider as my friends today. Vanessa & Benoit; Maarten, Begga & your interstellar daughter Sterre; Luc & Marjolein; Pieter & Jan; Joeri, Greet & Mona (she's a punk rocker); Youri & Melissa; Jürgen & Claudia; Mario & Fabien; Boris & Marika; Valère, Wim, many thanks for the numerous weekends, trips when somebody was staying abroad, travels around the world, skiing holidays, barbecues, Risk battles, parties in Ghent (where it seems there's always something to celebrate), and of course . . . Dranouter.

Working on a PhD equals to being seated for practically most of the day. So, as a result of wanting to get rid of my excess energy and to compensate for my sometimes Burgundian lifestyle, I found out that most of the other people I would like to thank can be associated (classified) with many of my after-work sports activities. *'t Jong Geweld*: Kurt, Kathleen & Femke; Pieter & Elke; Aaron 'Ronne' & Inge; Tom 'Coeman' & Kartrien; Bart 'de Witten' & Sarah; Tom 'Waldo' & Tine, Nele 'Poerk'; Maarten 'Co' & Lore; Joke; Willem 'Willy the King' & Katelijne; Pieter 'Coeman'. *ADW Platte Daken*: Stefan, Stephan, Joost, Hendrik 'Rikkie', Gert. The RVT volley team: Kris, Mieke, Chris, Eddy, Jan, Katrien, Dirk

'Boerie', Vero, Geert, Yves, Elke. Nonetheless, some might strongly argue that my *mens sana in corpore sana* creed is badly compensated by the social events that tend to come after the exercise.

The road of science is seeded with many interesting and intriguing people, and although it is slightly unfair to single out individuals, it would be even more unfair not to mention anyone. So I would like to especially mention some colleagues I've met along the way: Marnix Vandaele and Walter Bossaert, my two promotors during my studies of mathematics and computer science; Luc Vauterin, Paul Vauterin and Bruno Pot from the Applied Maths team; Mohammed Amar, thanks for hosting the workshop in Rabat and a wonderful week with David 'grosses frites' Smith, Micah Krichevsky, Gina Koenig and Philippe Desmeth; George Garrity, Juncai Ma, Hideaki Sugawara, François Guissart, Patricia Mergen, Jurgen Tack, Hendrik Segers, Paolo Romano, and all the people at BCCM, EBRCN and BEBIF.

To conclude with, I have to say a big "Thanks"to my brothers Tom & Lieselot, Jeroen & Ilke who stuck with me through so many adventures, and also to the other Marble members Stijn, Maarten, Geert and Arvid. And last but certainly not least, my most bountiful gratitude goes to my parents. Guido and Ingrid, the encouragement and care you gave me throughout all those years really goes beyond words.

And for all of you who were wondering where I have been hanging around lately, and whom I may have painfuly forgotten to mention above: be warned, because now the writing of this book is over, I'm back on the road again ...

Peter

Chapter 1

A Backpacker's Guide to the Landscape of Bacterial Taxonomy

"There is a theory which states that if ever anyone discovers exactly what the universe is for and why it is here, it will instantly disappear and be replaced by something even more bizarre and inexplicable.

There is another which states that this has already happened."

— Douglas Adams

WHAT are those wondrous life forms at the other side of the microscope? A question that has intrigued many scientists ever since the pioneering era when Antonie van Leeuwenhoek was first admiring these free-living bacterial cells through his 300 times magnifying single lens microscope. But even long before the discovery of microbes, scholars of Aristotle's school were trying to group the myriad of observed living organisms into natural and meaningful classes. This pursuit remains active, and the classifications are, to some degree, still controversial. Figure 1.1 shows the example of a very incomplete and informal *taxonomic tree*, inspired by the stratified subdivisions proposed by Woese, Kandler and Wheelis. This particular tree of life drills down the branch containing the mammals, leaving the other offshoot less specified. Bacterial subdivisions follow the same general taxonomic outline, only the names may seem somewhat less familiar. Traditionally, these classifications were based upon the morphology of organisms. Literally, morphology means shape, but it is generally regarded to include the internal structure as well. In this context, the particular genetic encoding for an organism is called its *genotype*, whereas the resulting set of physical characteristics of an organism is called its *phenotype*. Morphology is only one part of the phenotype, where other parts include physiology, or the function of living structures, and development. Nowadays, the life science taxonomies are

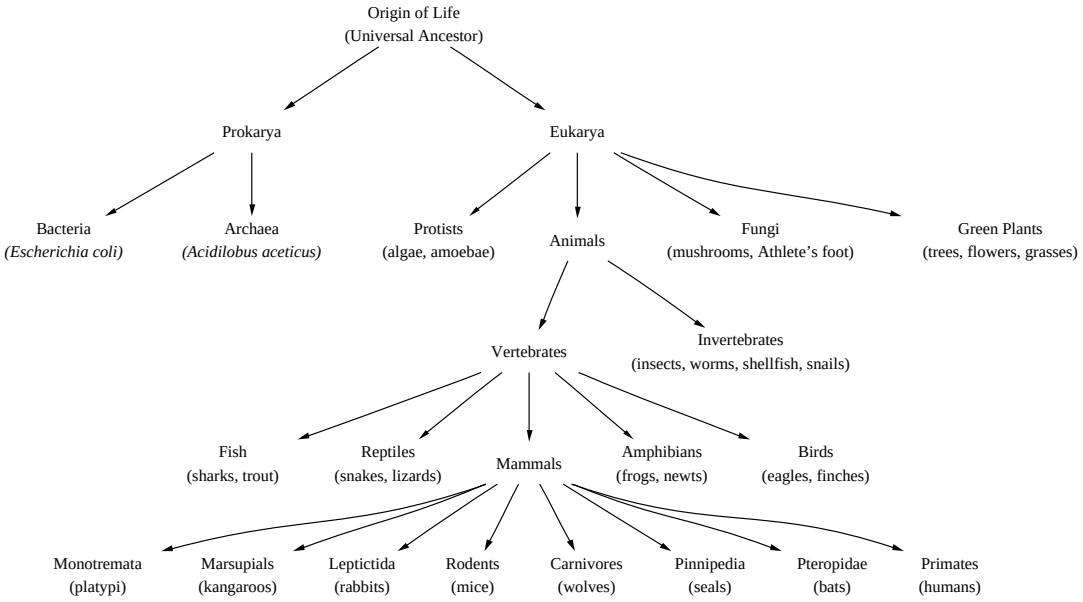


Figure 1.1: A very incomplete and informal taxonomic tree. Items between brackets are common or scientific names of representative organisms or classes.

increasingly tested upon and updated against the knowledge of molecular structures and sequences, which are generally regarded to be more revealing of evolutionary relationships than are classical phenotypes. Particularly so among microorganisms.

Bacterial evolution is a complex and dynamic machinery, where the creation of new combinations is performed in the genotypic search space, whereas selection of the most feasible specimen occurs at the phenotypic level, by means of the evaluation of a natural objective function known as *survival of the fittest*. This discrepancy between genotype and phenotype is very important because small allowable steps in the genotypic space may have large consequences at the phenotypic space. Natural evolution is responsible for the many spectacular abilities of living things, and for their tremendous diversity. Studying this diversity of microorganisms as they are observed today forms the main subject of research in *bacterial taxonomy*, whereas investigating the history of organismal lineages as they change through time is particularly called the domain of *phylogeny*.

Polyphasic bacterial taxonomy is aimed at the integration and processing of many kinds of data (phenotypic, genotypic and phylogenetic) on microorganisms, and essentially strives after the delineation of an objective consensus type of taxonomy that gives evidence of the least number of anomalies when confronted with all the collected empirical information. The generally accepted present-day *bacterial species concept* heavily relies on the definition of a threefold set of quantitative comparative rules, stating that the sample variability within a species is restricted to $\geq$ 97% 16S rRNA sequence similarity, $\geq$ 70% DNA-DNA homology and a $\leq$ 2% difference in the G+C fraction of the complete genome. This conceptual definition might seem quite arbitrary, while many issues, such as the frequency of

horizontal gene transfer occurrence and its impact on the existing classification schemes, remain open questions that challenge taxonomists today. Technical hurdles and a tedious manual approach taken for the integration of dozens of information sources have kept the current scope of most polyphasic studies rather minimal, whereas a great deal of subjective decision making for the derivation of a consensus view on the data is left on behalf of the microbiologist's personal interpretation. This turns the validation of the existing species definition against new empirical information into a slowly ongoing process.

To break the rigidity of this approach, however, one could envisage a global information system, which in a structured and uniform way captures the reams of experimental data that are generated in the field of microbiology. Such a knowledge managemental structuration would dramatically simplify the application of intelligent and well-founded data mining techniques, as tools for the discovery of objective and universal taxonomic consensus models in a more dynamic and a more automated manner. In addition, these automated reasoning systems could adapt in a flexible way to the advent of new incoming data and interact with the outside world whenever some of the necessary pieces for completion of the taxonomic puzzle are missing or unclear. As well as new insights and hypotheses on microbial life and its evolution could be easily tested against this vast knowledge base, and possibly have an instant impact on standing taxonomic models. All validly described taxa, their cultured and investigated strains, raw empirical information, published research documents, and scientific names assigned to concepts during previous research should get their place in the information system. Multiple cross-reference links should connect the related pieces of information in the knowledge base. As such, new experimental data fed to the integrated system could lead to an automatic re-evaluation of the existing taxonomy, whereas the improved synthesis would become instantly available for the scientific community. The old-fashioned style of manually composing the rigid Linnean descriptions, which are still part of the routine taxonomic practice, would become completely obsolete, making the taxonomist's job much more challenging as the process of boring repetitive descriptions is basically taken over by the information system. As a result, the taxonomist could focus on resolving the more interesting evolutionary and other exiting fundamental biological issues.

This thesis is an attempt to bridge just that range of exploration, from raw data to abstract concept, or from practice to theory, in contemporary microbial taxonomy. As a result, it is situated on the cross-roads between microbiology, mathematics and computer science. The art of drawing the landscape of bacterial diversity, used as figure of speech for taxonomic modeling, is conceptualized by the three pendicular spatial axes depicted in Figure 1.2, which generally correspond to the different domains of science that may contribute to resolve the problem: measuring a representative variety of reproducible and comparable experimental features on sets of bacteria (microbiology/taxonomy), designing objective classification methods for finding groups in data in an unsupervised way (mathematics/classification) and combining all experimental data and their different groupings in a uniform and intelligent way (computer science/knowledge management). Current progress in taxonomic modeling of the bacterial diversity has often restrictively exploited only one or two of these dimensions at the same time. However, the axes are osmotically intertwined within the envisaged microbial information system.

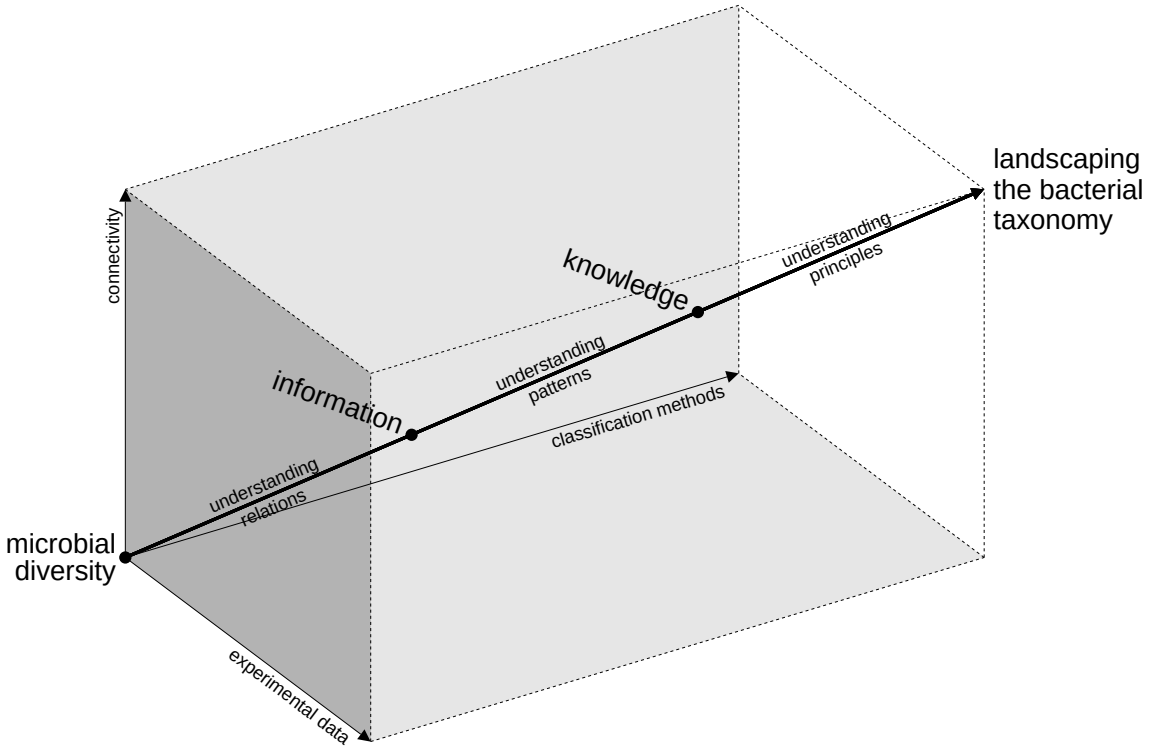


Figure 1.2: Learning curve for understanding and modeling of a taxonomy that fits as closely as possible to the observed phenomena of bacterial diversification.

With the explosion of recorded information, microbiologists for the first time found it necessary to familiarize themselves with databases and the algorithms needed to extract the correlations between records, and in turn have put these to good use in the exploration of natural relationships. This has emerged an ever-increasing environment of heterogeneous and autonomous data sources, providing partially overlapping knowledge on investigated microorganisms. A basic effort of the landscaping activity henceforth is devoted to the construction of solid pathways and bridges, which connect the related pieces of information that are scattered throughout the scenery. As a result, microbiology might seriously benefit from the design of intelligent software agents that can assist in the navigation through this bacterial countryside, together with the development of data mining tools that can aid in the discovery of new relations, patterns and principles that describe the steering mechanisms of environmental evolution. An important barrier for understanding biology is learning its language, which needs to be unequivocally defined before it can be correctly interpreted by self-learning information systems. To this end, chapter 2 introduces the implementation of an *integrated strain database*, a central repository aimed at the complete and correct accumulation of the strain label equivalence relation used for referencing cultured bacterial samples into a wide variety of data sources. Traditionally, the strain label synonym information has been fragmentarily disseminated by a number of autonomous data providers, and suffers heavily from syntactical variation, ambiguities and other inconsistencies. Through the establishment of a solid framework for resolving these issues, the integrated strain database might constitute the cornerstone of a divide and conquer strategy for the management of

distributed microbial information, seamlessly glueing together the different pieces of the taxonomic puzzle.

Primordial for recording and getting familiar with the diverse bacterial landscape is the compilation of clarifying maps. Founded on the widely accepted Darwinian theory of evolution, which states that any pair of organisms, no matter how different, has a common ancestor somewhere in the near or distant past, stratified representations have since long been the traditional instrument employed by many taxonomists to perform their cartography of the microbial diversity. Applications range from the development of complete taxonomies, to the delineation of the different subspecies of a distinct but varied species. The ability to draw hierarchies as a means to model the naturally occurring relationships between a set of bacterial samples on the basis of their empirically determined features, is intimately coupled to the characteristic min-transitivity of the pairwise similarity matrices, which are calculated from the choice of a similarity coefficient that makes an estimation of the relatedness between each pair of strains. However, similarity matrices derived from the experimental bacterial characteristics are generally not min-transitive by nature. This has led to the development of a large battery of *hierarchical clustering methods*, that approximate the given experimental similarity matrices by closely neighbouring similarity models that truely are min-transitive. Chapter 3 situates some of the traditional hierarchical clustering algorithms into the general mathematical framework of transitive openings, closures and approximations, and discusses some newly developed algorithms that belong to the same family. The relative merits and demerits of this wealth of techniques are examined through a series of comparative experiments.

Just as every backpacker trusts on his guidebook that contains several colorful maps, each highlighting some other aspects of the surrounding countryside or representing the scenery with a different level of detail, there might be many meaningful groupings for a given set of microbial features, each reflecting different insights into the underlying natural relationships of the strains. Consequently, if there are several meaningful groupings, a variety of classification techniques could be needed to reveal them all. However, taxonomists traditionally have merely relied solely on hierarchical clustering methods to unravel natural relationships among microorganisms. An approach that might result in a distorted view on the multifaceted bacterial landscape. It is particularly appreciated that when using some hierarchical method of clustering analysis, early decisions in the construction process may preclude certain meaningful groupings at later stages. This unilateral kind of analysis thus completely ignores the existence of state-of-the-art non-hierarchical classification methods for learning the hidden relationships behind sets of bacterial features. Chapter 4 has sought to shatter this tradition for the particular case of classifying microbial strains on the basis of their genotyping fingerprint profiles, an experimental technique used to sample the bacterial genome that results in highly specific banding patterns. Applying classification methods for the analysis of genotyping fingerprint patterns often requires several preliminary transformations of the original data representation into a more workable computational format. It is shown that a naive choice of the discretization method for turning molecular banding patterns into binary vector format can have a harmful impact on the final classification of the profiles. This has led us into an evaluation of the existing multiple band matching methods and the introduction of a new technique – called *sliding window*

discretization – for transforming genotypic fingerprinting data into binary vector format. In the context of an extensive set of fluorescent amplified fragment length polymorphism (fAFLP) fingerprint patterns from strains of the family *Vibrionaceae*, it was demonstrated that sliding window discretization results in the most lossless vector transformation compared to other methods. Accordingly, the binary vectors were classified according to the *minimization of stochastic complexity*, as an alternative strategy for the hierarchical clustering algorithms that is based on the optimization of an information theoretic expression. A scrutinized comparison of the classifications for the same set of fAFLP fingerprint patterns by different classification strategies has revealed that there was good overall correspondence between the alternative groupings, but also confirmed that no single classification managed to reflect all the taxonomic relationships within the *Vibrionaceae*. The question whether a single roadmap/taxonomy can be distilled that encompasses all aspects learned on bacterial diversity, however, remains open.

Once all locations in the scenery are sufficiently accessible and connected, detailed maps are drawn up to guide navigation, and our backpacks are stuffed with suitable camping equipment, the discovery of new patterns can begin by means of a systematic exploration of the landscape. With this in mind, chapter 5 attempts to demonstrate the implications and abilities of knowledge discovery in databases as a valuable technology for the evaluation of massive quantities of genotypic and phenotypic features acquired on microorganisms. This all-round analysis strategy aims at the application of a broad spectrum of data mining tools, bearing in mind the necessity of preliminary steps in data cleansing, integration and warehousing. In particular, exploitation of the vast amounts of information accumulated during fifteen years of routine gas chromatographic analysis on the fatty acid content of environmental aerobic bacteria is brought into focus. Along the lines of this investigation, it is shown how learning new information for enhancing the discriminatory power of an automated fatty acid recognition system, itself may gradually improve the resolution of the technique for bacterial classification and identification, especially for some species that previously were undistinguished by the methodology.

Achieving the goals of a self-learning reasoning system for landscaping the bacterial taxonomy, yet means that several major technical and organisational hurdles will need to be overcome. This includes advancing the barriers of global data sharing, identify and come up with ways to fill the gaps of observational efforts, and explore the possibilities of novel data mining techniques to the benefits of understanding bacterial life. Despite the slew of unresolved issues, let's put on our hiking shoes and hit the road. . .

Chapter 2

The content of this chapter is a strongly based on the published or submitted material in the following scientific journal papers:

[1] **Dawyndt, P., Vancanneyt, M. & Swings, J. (2004).** On the integration of microbial information. Sugawara H. (ed.). *WFCC Newsletter* **38**, 19–34, World Federation for Culture Collections.

[2] **Dawyndt, P., Vancanneyt, M., De Meyer, H. & Swings, J. (submitted).** Knowledge accumulation and resolution of data inconsistencies during the integration of microbial information sources. *IEEE Transactions on Knowledge and Data Engineering*.

Chapter 2

Knowledge Accumulation and Resolution of Data Inconsistencies during the Integration of Microbial Information Sources

”Real–world data is dirty”

— Hernández, M. A. & Stolfo, S. J.

THE internet has emerged as an ever-increasing environment of multiple heterogeneous and autonomous data sources that contain relevant but overlapping information on microorganisms. Microbiologists might therefore seriously benefit from the design of intelligent software agents that assist in the navigation through this information-rich environment, together with the development of data mining tools that can aid in the discovery of new information. These applications heavily depend upon well-conditioned data samples that are correlated with multiple information sources, hence accurate database merging operations are desirable.

Information systems designed for joining the related knowledge provided by different microbial data sources are hampered by the labelling mechanism for referencing microbial strains and cultures, that suffers from syntactical variation in the practical usage of the labels, whereas additionally synonymy and homonymy are also known to exist amongst the labels. This situation is even complicated by the observation that the label equivalence knowledge is itself fragmentarily recorded over several data sources which can be suspected of providing information that might be both incomplete and incorrect.

This chapter presents how extraction and integration of label equivalence information from several distributed data sources has led to the construction of a so-called integrated strain database, which helps to resolve most of the above problems. Given the fact that information retrieved from autonomous resources might be overlapping, incomplete and incorrect, much energy was spent into the completion of missing information, the discovery of new associations between information objects and the development and application of tools for error detection and correction. Through a thorough evaluation of the different levels of incompleteness and incorrectness encountered within the incorporated data sources, we have finally given proof of the added value of the integrated strain database as a necessary service provider for the seamless integration of microbial information sources.

2.1 Introduction

Polyphasic taxonomy [71] of microorganisms is aiming at the integration and processing of many different kinds of data and information (phenotypic, genotypic and phylogenetic), and essentially strives after the delineation of an objective consensus type of taxonomy that represents the least amount of anomalies when confronted with all the attained knowledge. Technical difficulties and a time-consuming manual approach to the integration of dozens of information sources have kept the current scope of most polyphasic studies rather minimal, whereas a great deal of subjective decision making for the derivation of a consensus view on the data is left on behalf of the microbiologists. One could however envisage a global information system that in a uniform way captures the reams of data generated in the field of microbiology. This data structuration process would enable the intelligent application of well-founded data mining techniques as a means to discover objective and universal taxonomic consensus models on a more dynamic and a more automated manner. Such a system could then adapt in a flexible way to the advent of new incoming data and interact with the outside world when necessary pieces for the completion of the taxonomic puzzle are missing or unclear.

A major technical hurdle that must be overcome for achieving this goal, is to concisely join all related information provided by the many distributed databases that contain relevant information on microorganisms. This general problem of merging multiple databases acquired from different sources with heterogeneous representations of the information is frequently encountered in knowledge discovery in databases (KDD) and decision support applications of large commercial and governmental organizations [37], but also finds its way into several data-driven branches of science [26, 62], including biology [8, 16]. The database integration process is difficult to solve both in scale and accuracy, and features many aspects that need to be tackled. Notice however that more (and more reliable) conclusions can generally be drawn from the databases, after the integration process has been conducted. Moreover, without performing the necessary data cleaning steps during the process of database integration, many of the data mining algorithms that can be applied would be rendered useless, as they heavily depend on the quality of the data under investigation [57].

With the rapid emergence of data formats and applications in bioinformatics supporting a veritable cottage industry of databases, the design of commonly accepted and implemented data formats and interrogation languages becomes paramount to support holistic scenarios [62]. The issue of querying databases in environments where the distributed data sources have different schemas has been addressed extensively in literature, and is known as the *schema integration* problem [4, 44, 52, 54]. Multiple common schema design initiatives for the standardization of data exchange between distributed microbial data providers have arisen over the past two decades: Microbial Information Network Europe (MINE; [30, 70]) and Common Access to Biotechnological Resources and Information (CABRI; [9]) are standard schemas designed specifically for disseminating information on microorganisms, while the Global Biodiversity Information Facility (GBIF) supports both Access to Biological Collection Data (ABCD; [1]) and Darwin Core [15] as standard schemas to cover all information about the complete biodiversity on earth.

Successful database integration does however not only require the development of common schemas which allow searching the different information sources from a logical single point of access, but also urges that the collected information is normalized and corrected wherever necessary. In the framework of this chapter we are therefore primarily interested in resolving the duplications and other inconsistencies showing up on the data level of distributed microbial information sources. These *data integration* issues are complementary to their schema integration counterparts, but do not seem to have been fully addressed within the problem domain of microbiology. One particular inconsistency that occurs on the data level is the existence of multiple representations for the same real-world entity within environments lacking unique consistently-used identifiers for the different object instances [13]. This forms a serious hindrance for extracting all the related information on a given entity from multiple distributed data sources. Currently there exists for example no systematic cross-referencing between resources that supply basic strain information (see Table 2.6 for some examples) and other public data sources such as nucleotide and protein sequence databases [40] and scientific literature databases that provide additional features of the strains. Automatic execution of basic computational operations for microbial applications, such as collecting all the 16S rDNA sequences of the *Enterococcus* spp. type strains deposited in the public sequence databases, may as a result turn out to be very complicated when the request must be completely and correctly answered on the fly [16]. Solutions for this problem appearing in the literature have been called *record linkage* [27, 59, 60], *duplicate record elimination* [5], the *inter-database instance identification* problem [72], *hardening soft databases* [13], the *data cleaning* (or *data cleansing*) problem [26] or the *merge\purge* problem [36, 37]. Most deduplication techniques restrict themselves to the detection and resolution of syntactical differences between the descriptors of the same entity. The labelling mechanism applied for referencing microbial strains and cultures, however, does not only suffer from syntactical variations in its practical usage, but additional semantic issues such as synonymy and homonymy are also known to exist amongst the strain numbers. The problems for this specific research area are even more complicated by the observation that the semantic label equivalence information is itself fragmentarily recorded over several data sources, which can be suspected of providing information that might be both incomplete and incorrect.

The urgent need to get a complete and correct view on the semantic label equivalences that exist in the field of microbiology, as required prior knowledge for setting up more intelligent information systems, has inspired us into the construction of a so-called *integrated strain database*. In section 2.2 we explain in detail how the basic label equivalence information can be extracted and integrated from several distributed and heterogeneous data sources, in order to achieve a concise and complete representation of the equivalence knowledge within the integrated strain database. This process both supports the completion of missing information and the discovery of new associations between information objects. The information captured within the integrated strain database is however seriously threatened by the fact that the quality of information retrieved from autonomous resources might be poor and that overlapping information can be contradictory. Section 2.3 therefore presents an arsenal of error detection and correction tools that we have developed and applied as a means to improve the overal correctness of both the integrated strain database and its composing primary data sources. Some basic properties and statistics of the current version of the integrated strain database are reviewed in section 2.4, together with a thorough evaluation of the different levels of incompleteness and incorrectness encountered within the currently incorporated data sources. These latter quality estimations should endorse the added value of the integrated strain database as a necessary service provider for the seamless integration of microbial information sources. Section 2.5 further works out on this, by demonstrating how the integrated strain database forms the cornerstone of a solid and manageable cross-referencing system that establishes mutual links between the information provided by biological resource centers, emperical knowledge bases and scientific research papers.

2.2 Construction of an integrated strain database

Data supplied by autonomous data sources typically include identifiers for real-world entities that may vary among the different data sets, due to a wide variety of reasons. Hence, the equality of two values over the domain of a common join attribute is not specified as a simple arithmetic predicate, but rather by a series of equational axioms that define equivalence, i.e. by an equational theory [36, 37]. We start this section with an informal description of several features of the equational theory that corresponds with the labelling mechanism that is commonly applied to refer to strains and cultures in the field of microbiology. The observation that homonymous strain numbers exist implies that these labels cannot be applied as unique identifiers for the unequivocal discrimination of strains and cultures in a global information system. Moreover, as the semantic equivalence of synonymous strain numbers is related to the mathematical notion of an equivalence relation, it should always respect the properties of reflexivity, symmetry and transitivity. The practical definition of the synonym equivalences, however, is currently dispersed over a myriad of independent information sources, so that the semantic transitivity of the equational theory is frequently broken and not completely accessible through a single point of entry. In a second part of this section we therefore discuss how a central repository can be evoked, which simultaneously resolves the ambiguities of homonymous strain numbers and offers a complete picture on the equivalence relation of synonymous strain numbers.

2.2.1 Equational theory for the microbial labelling system

Any method for learning and detection of duplicate representations and occurrences of the same real-world entity within a given context, needs an underlying theory that defines how the equivalence of two different representations is determined. This is refered to as the *equational theory* or the *similarity model* of the problem domain, and it is said that the equational theory dictates the logic of domain equivalence. The process of creating a good equational theory is similar to the process of creating a good knowledge-base for an expert system. Hence, in order to give a formal description of the similarity model for complex problem domains, an expert that is intimately familiar with the context is needed. After all, any improperly defined equational theory will lead to either an increase in the number of falsely matched entities (false positives) or to a decrease in the number of matching descriptors that represent the same entity (false negatives) and accordingly should have been recognized by the similarity model as being equivalent [37].

We set off with a rather informal introduction on the different concepts of an equational theory for determining the equivalence of the strain numbers that are used as descriptors for cultures and strains in the context of microbiology. Later on, in subsection 2.2.2, we will come to a more formal explanation of how this similarity model was implemented for the construction of the integrated microbial strain database, which is the subject of discussion of this chapter. Staley and Krieg [69] describe the concepts of (bacterial) strains and (bacterial) cultures in the following way. A (bacterial) *strain* is made up of the descendants of a single isolation in pure culture, and usually is made up of a succession of cultures ultimately derived from an initial single colony. They see a (bacterial) *culture* as a population of (bacterial) cells of the strain, instantiated in a given place during a given time, e.g. in a test tube, on an agar plate or in a cryopreserved or lyophilized state intended for long preservation. Although these formalisms have been specifically described here for the microbial subdomain of bacteriology, they are equally applied for other kinds of microorganisms, such as fungi and yeasts. Dijkshoorn *et al.* [21] make further distinction between a *strain in the taxonomical sense* and a *strain in nature*, but we will stick to the unqualified strain terminology, implicitly referring to the former epithet. Different *labels* in the form of textual strings (commonly called *strain numbers* by microbiologists) or barcodes are assigned to some of the cultures, for the purpose of encoding properties of that specific culture in a notebook or a database. Whether the occurrence of a label should be interpreted as a reference to a specific culture in the strain history or whether it is simply used as an exemplar for referring to a strain as a whole, may depend on the context wherein the label is used. In order to avoid confusion with the abstract strain concept as it is formalized within the equational theory described here, we have therefore opted in favor of the more general term label further on in this chapter, instead of sticking to the more commonly applied strain number terminology used in the field of microbiology.

Labels that reflect the same real-world entity and for which the equivalence can be derived purely based on the syntactical representation of the labels, are called *syntactically equivalent* labels. Minor variations in the spelling of a label assigned to a microbial culture might occur, due to the fact that strict formatting rules for labels are lacking or not adhered to in practical situations. With this in mind, the concept of a culture

was abstracted in the integrated strain database as the equivalence class of all syntactical variations encountered within the practical occurrences of the label assigned to a culture, whereby the syntactical equivalence of two labels can be evaluated by a matching function that allows for small variations in the spelling of the labels. Measurement of the syntactical distance (or edit distance) between two string representations has been the subject of a largely studied research area known as *approximate string matching*, from which a vast amount of domain-independent approximate string matching algorithms have emerged [6, 12, 23, 28, 34, 45, 47, 50, 58, 61, 66]. Based on the domain-specific knowledge of an entity and its semantics in the given problem domain, an interpretation function for the evaluation of the syntactical equivalence of two entity labels may then be moulded from the stipulation of a series of production rules, which can make use of approximative string matching functions as their basic building blocks. Numerous examples of syntactical equational theories that are implemented accordingly, have been published for a wide variety of concepts such as customer addresses [2], web pages [7], scientific publications [39, 55], lexicon variants [41], medical patients [59, 60], fraud and money laundry actors [64], census data [65] and business documents [75], amongst many others.

Management of syntactical equivalences in the integrated strain database has been organised on the basis of the textual decomposition of labels into the three syntactical units that are schematically depicted in (2.1).

$$< \texttt{label} > := < \texttt{acronym} >< \texttt{index} >< \texttt{postfix} > . \tag{2.1}$$

The *acronym* of the label is composed of the prefix of the string representation of the label up to (but not including) the first numerical character. The continuous substring of numerical characters that follows the acronym constitutes the label *index*, whereas the remaining part of the string is called the *postfix* of the label. This decomposition follows the common practice in microbiology that most labels are constructed by appending a descriptive string for the instance that has assigned the label (e.g. the surname of an individual researcher that has isolated the strain, the acronym of a culture collection that has received a sample of the strain or the acronym of a scientific project wherein the strain was applied or investigated) with a unique numerical identifier that discriminates all the cultures worked with by that instance. Additionally, different mechanisms are in place for the indication of further subcultures of a given culture, which generally have in common that they add some textual string at the end of an existing label. These additions are captured within the postfix unit of the label decomposition. Within the decomposed representation of the labels, one or more units are allowed to represent empty strings. The integrated strain database then automatically tackles most syntactical variation confronted with during practical usage of the labels, by the implementation of a series of production rules which transform the decomposed units of a given label into a so-called *normalized syntactical form*. Some production rules for example trim the acronym and postfix units of a label from trailing and leading white space characters and other delimiters such - or _ characters, or remove the trailing T symbol of the postfix, which is merely an indication that the label corresponds with a type strain, rather than being a real part of the label itself. Syntactical equivalence of two labels can then simply be determined by checking whether both labels have the same normalized syntactical form. Acronym and postfix comparisons are performed case-insensitive, whereas the equality of indexes is irrespective of the number of leading zeroes.

The integrated strain database also uses the decomposition of labels into their normalized syntactical form, for implementing a mechanism to standardize the format of all labels assigned by the same instance. This mechanism is granular in that different formatting styles may be set depending on the acronym of the label. The integrated strain database controls the display style of the labels by the assignment of a customizable template format to each acronym. These formatting templates for example indicate the string delimiters that must be uniformly used between the acronym, index and postfix units of all labels having a given acronym, and discriminate the acronyms that require leading zeroes to attain fixed-length indexes from the acronyms that use variable-length indexes.

Equivalence of the labels used for referencing strains and cultures in the field of microbiology can however not be derived solely from the typography of the labels. Labels which semantically refer to the same real-world entity but differ substantially in their syntax, are generally named *aliases* or *synonyms*. This kind of equivalences has to be defined on the basis of more axiomatic foundations. It is common practice in microbiology that most instances growing microbial cultures (either individual researchers, culture collections, or research and industrial laboratories) apply their own system for labelling the biological samples they work with, whereas these labels are considered to be unique identifiers for encoding information in the context of that instance. When a sample of biological material gets transfered from one instance to another, the receiving party may assign a new synonym label to the received culture according to its own labelling system and communicate this new alias to the depositor for reasons of traceability. This relabelling tradition is mainly kept alive for proper discrimination of both cultures harboured by the depositor and the receiver, as a means to implement a total data quality management (TDQM) system for monitoring the distribution of microbial strains and tracing possible contaminations. In this way, as more cultures of a single isolate in pure culture get distributed over numerous instances, the amount of different labels that refer to the same microbial strain grows accordingly. Moreover, data generated from and properties encoded on a given strain by different people at different locations get completely defragmented, because in most cases reference is made to the biological material using only the label of the culture that was worked with. This observation forms a serious hindrance for the retrieval of all known information about a given strain, in a sense irrespective of whatever synonym label that was used for referencing the strain. In order to avoid such Babel-like confusion, it is therefore essential to have the knowledge about all synonym labels that were assigned to any culture that descends from the same single isolate in pure culture. In this respect, the strain concept was abstracted in the integrated strain database as the equivalence relation of all cultures descending from a single isolation in pure culture, hence by transitivity also as the equivalence relation of all labels assigned to any of these cultures, with a built-in syntactical relaxation on the spelling of the labels.

At present, alias information of the microbial labelling system is fragmentarily recorded over a number of laboratory notebooks, scientific papers that describe new microbial isolates or reuse the biological material of previous studies, and catalogues of instances that harbour cultured samples for further dissemination. Additionaly, enumerations of important and well-documented strains (primarily type strains) are regularly recompiled from information extracted from the above resources [31, 73], for the benefit of a readership that

data source	record ID	species name	synonym labels
ATCC	ATCC 14579^{T}	*Bacillus cereus*	971; 13; NCIB 9373; NCTC 2599
CABRI	CIP 66.24^{T}	*Bacillus cereus*	ATCC 14579; CCM 2010; DSM 31; IAM 12605; JCM 2152; LMG 6923; NCIB 9373; NCTC 2599
CABRI	DSM 31^{T}	*Bacillus cereus*	ATCC 14579; CCM 2010; LMG 6923; NCIB 9373; NCTC 2599
CABRI	LMD 75.8^{T}	*Bacillus cereus*	ATCC 14579; NCIB 9373; NCTC 2599; CCM 2010; DSM 31; Gibson
CABRI	LMG 6923^{T}	*Bacillus cereus*	ATCC 14579; CCEB 625; CCM 2010; CCUG 7414; CECT 148; CIP 66.24; DSM 31; FIRDI 603; Ford 13; Gibson 971; IAM 12605; JCM 2152; Logan B0002; NCFB 1771; NCIB 9373; NCTC 2599; NRRL B-3711; OUT 8406
CABRI	NCIMB 9373^{T}	*Bacillus cereus*	ATCC14579; CCM2010; CECT148; CIP66.24; DSM31; IAM12605; IFO15305; JCM2152
CCM	CCM 2010^{T}	*Bacillus cereus*	ATCC 14579
CCRC	CCRC 10603^{T}	*Bacillus cereus*	ATCC 14579; CCM 2010; DSM 31; NCIMB 9373; NCTC 2599
CCRC	CCRC 11026^{T}	*Bacillus cereus*	IAM 12605
CCUG	CCUG 7414^{T}	*Bacillus cereus*	CCM 2010; NCIB 9373; ATCC 14579; NCTC 2599; Ford 13; DSM 31
CECT	CECT 148^{T}	*Bacillus cereus*	ATCC 14579; CCEB 625; CCM 2010; CCRC 10603; CCRC 11026; CCTM La 3674; CCUG 7414; CIP 66.24; DSM 31; FIRDI 603; Ford 13; Gibson 971; IAM 12605; JCM 2152; LMD 75.8; LMG 6923; NCFB 1771; NCIMB 9373; NCTC 2599; OUT 8406; VTT E-93143
CECT	CECT 5050^{T}	*Bacillus cereus*	ATCC 14579; CCM 2010; CECT 148; DSM 31; Ford 13; Gibson 971; LMG 6923; NCIMB 9373; NCTC 2599
CIP	CIP 66.24^{T}	*Bacillus cereus*	ATCC 14579; CCM 2010; DSM 31; IAM 12605; JCM 2152; LMG 6923; NCTC 2599; NCIMB 9373
DSMZ	DSM 31^{T}	*Bacillus cereus*	ATCC 14579; CCM 2010; LMG 6923; NCIB 9373; NCTC 2599
IFO	IFO 15305^{T}	*Bacillus cereus*	ATCC 14579; CCM 2010; DSM 31; IAM 12605; JCM 2152; LMG 6923; NCIB 9373; NCIMB 9373; NCTC 2599
JCM	JCM 2152^{T}	*Bacillus cereus*	ATCC 14579; CCM 2010; CCRC 10603; CCUG 7414; CECT 148; CIP 66.24; DSM 31; IAM 12605; IFO 15305; KCTC 3624; LMG 6923; NBRC 15305; NCFB 1771; NCIMB 9373; NCTC 2599; NRRL B-3711; VKM B-504
KCTC	KCTC 3624^{T}	*Bacillus cereus*	ATCC 14579; CCM 2010; CCRC 10603; CIP 66.24; DSM 31; IAM 12605; IFO 15305; JCM 2152; LMG 6923; NCFB 1771; NCIMB 9373; NCTC 2599; VKM B-504
LMG	LMG 6923^{T}	*Bacillus cereus*	ATCC 14579; CCEB 625; CCM 2010; CCUG 7414; CECT 148; CIP 66.24; DSM 31; FIRDI 603; Ford 13; Gibson 971; IAM 12605; JCM 2152; Logan B0002; NCFB 1771; NCIB 9373; NCTC 2599; NRRL B-3711; OUT 8406
NRRL	NRRL B-3711^{T}	*Bacillus cereus*	ATCC 14579; DSM 31; NCIB 9373; NCTC 2599
UKNCC	NCTC 2599^{T}	*Bacillus cereus*	Ford 13; ATCC 14579; NCIB 9373; DSM 31
UKNCC	NCIMB 9373^{T}	*Bacillus cereus*	Gibson971; Ford13; ATCC14579; CCM2010; CECT148; CIP66.24; IAM12605; IFO15305; JCM2152; NCTC2599; NCDO1771
taxa (DSM)	type strain	*Bacillus cereus*	ATCC 14579, CCM 2010, DSM 31, NCIB 9373, NCTC 2599
integrated strain database	type strain SID = 1101	*Bacillus cereus*	ATCC 14579; CCEB 625; CCM 2010; CCRC 10603; CCRC 11026; CCTM La 3674; CCUG 7414; CECT 148; CECT 5050; CIP 66.24; DSM 31; FIRDI 603; Ford 13; Gibson 971; Gibson; IAM 12605; IFO 15305; JCM 2152; KCTC 3624; LMD 75.8; LMG 6923; Logan B0002; NBRC 15305; NCDO 1771; NCFB 1771; NCIB 9373; NCIMB 9373; NCTC 2599; NRRL B-3711; OUT 8406; VKM B-504; VTT E-93143; 13; 971

Table 2.1: Excerpt of the synonym equivalence information for the *Bacillus cereus* type strain, retrieved from several online resources. The last row of the table shows a normalized representation of the search results, where complete deduplication of the synonym labels assigned to the *Bacillus cereus* type strain was performed with the help of the `IncrementEquivalence` procedure discussed in subsection 2.2.2. See Table 2.6 for details on the different data sources.

is interested in reusing these strains for a variety of purposes. Typically, the authors of these summarizing documents have collected a shortlist of synonym labels for the listed strains, as an indication of the different instances from which cultures of the strains can be retrieved. In the remaining part of this chapter we will simply refer to any of the previously described information providers of synonymous labels as a *data source*. Manual extraction and collection of the complete synonym information from all of these data sources has proven to be a time-consuming and error-prone activity. As an example, the upper part of Table 2.1 shows an excerpt of the data sources that provide information on the synonym labels assigned to the *Bacillus cereus* type strain. This table clearly demonstrates that joining alias information from a bunch of different data sources, ends up with a great deal of duplication within the collected data. Calculation of the union of all the gathered information results in a normalized represention of all the known synonym labels, as is shown in the last row of the table. We will go deeper into an automated procedure for synonym information extraction and normalization in subsection 2.2.2. For the time being, we restrict ourselves to the observation that none of the contacted data sources contains complete information

SID	CID	label	species name	synonym labels
113562	368362	B2$^{\mathsf{T}}$	*Acinetobacter baylyi*	CIP 107474, DSM 14961
50260	268815	B-2	*Actinomadura madurae*	A 124, DSM 43381, IMET 7144, IMRU 1136
58752	347460	B2	*Aeromonas hydrophila* subsp. *hydrophila*	NCIMB 640
60298	345118	B2	*Bacillus* sp.	NCIMB 10936
20512	65830	B2$^{\mathsf{T}}$	*Chryseobacterium defluvii*	CCUG 47675, CIP 107207, DSM 14219
59765	350023	B2	*Clostridium butyricum*	KCTC 1902, NCIMB 9575
37398	267132	B-2	*Corynebacterium glutamicum*	ATCC 21269, KCTC 9853
40698	267459	B2	*Curtobacterium flaccumfaciens*	ATCC 33802
60975	303025	B2$^{\mathsf{T}}$	*Methylocapsa acidiphila*	DSM 13967, NCIMB 13765
43071	267727	B2	*Morganella morganii* subsp. *sibonii*	ATCC 51596
55683	269342	B-2	*Neisseria gonorrhoeae*	CCUG 13573
58441	346764	B-2	"Other unnamed bacteria"	NCIMB 17
48170	268485	B2	*Pseudomonas putida*	DSM 6376
35123	266888	B2	*Streptococcus salivarius*	ATCC 9759
9856	267562	B2	*Tenacibaculum maritimum*	ATCC 43397, CCM 3965, CECT 4276, CIP 103529, IAM 14118, IFO 16015, JCM 8137, LMG 10398, LMG 11611, NCIMB 2153, NCMB 2153, strain B2, Wakabayashi B-2

Table 2.2: Strain classes found within the current version of the integrated strain database, having cultures marked with label B2, or any syntactical equivalent label according to the equational theory defined in subsection 2.2.1. This list proves the usage of homonymous labels for indicating strains and cultures in the field of microbiology.

about all the synonym labels assigned to the *Bacillus cereus* type strain, whereas later on in this chapter (see section 2.4) we will demonstrate that this is not at all a standalone case.

Yet another problem concerning the labels used in microbiology that cannot be resolved on the syntactical level alone, is the context-dependency that might influence the semantic interpretation of the labels. Labels which are syntactically equivalent under the conditions of a well-defined syntactical equational theory, but represent different real-world entities depending on the context they are extracted from, are called *homonyms* [13]. Those labels that are considered to have no homonyms within the problem domain are called *unique labels*. Unique labels always have the same meaning, irrespective of the context they are used in, what distincts them from the so-called *ambiguous labels* that need an extra evaluation of the context before their exact semantical interpretation can be derived. Table 2.2 cleary demonstrates that we have to take into account the possiblity of homonymy amongst the labels that denote strains and cultures in the field of microbiology. This table shows a number of strains that all include cultures that have been tagged with the syntactical equivalent labels B2, B2$^{\mathsf{T}}$ or B-2, notwithstanding the fact that the context indicates that the addressed strains and cultures are not the same. Indeed, it is immediately evident from the table that the label B2$^{\mathsf{T}}$ is for example equivocally used as a reference to cultures (more specifically to the isolates) of the type strains of *Acinetobacter baylyi* [10], *Chryseobacterium defluvii* [43] and *Methylocapsa acidiphila* [17], apart from being assigned to cultures of a series of other strains. The problem of deriving the exact semantic interpretation of a label from an additional evaluation of the context from which the label was taken, has been intensively studied in the field of computational linguistics [63].

2.2.2 Algorithm for incremental learning of label equivalences

In the previous subsection we gave an informal description of the equational theory for the labelling system used in the field of microbiology. A formal similarity model for this problem domain should thus account for syntactical variations in the practical use of

the labels assigned to microbial cultures, and should in addition be able to cope with the homonymy and the synonymy of the labels, which are non-syntactical contributions to the semantic equivalences encountered within this problem domain. Let $\hat{\mathcal{L}}$ be the collection of all labels that refer to strains and cultures in the field of microbiology, so that we can write that

$$\hat{\mathcal{L}} \;=\; [l_1, l_2, \ldots, l_{|\hat{\mathcal{L}}|}]\,. \tag{2.2}$$

Due to the possibility of homonymy amongst labels in $\hat{\mathcal{L}}$, this collection must be captured within a *multiset* or a *bag* data structure when it is captured within an information system. This is reflected by the use of square brackets in the formulation of (2.2). Also remark that throughout this chapter we use the notation $|A|$ for indicating the size of a given countable collection A. The partition of $\hat{\mathcal{L}}$ that lumps together all labels associated to the same culture, is denoted as the countable set

$$\hat{\mathcal{C}} \;=\; \{\hat{c}_1, \hat{c}_2, \ldots, \hat{c}_{|\hat{\mathcal{C}}|}\}\,, \tag{2.3}$$

$$\text{where } \begin{cases} \hat{c}_i \subset \hat{\mathcal{L}} & 1 \leq i \leq |\hat{\mathcal{C}}| \\ \hat{c}_i \cap \hat{c}_j = \emptyset & 1 \leq i, j \leq |\hat{\mathcal{C}}|,\ i \neq j \\ \bigcup_{i=1}^{|\hat{\mathcal{C}}|} \hat{c}_i = \hat{\mathcal{L}}\,, \end{cases}$$

and each class $\hat{c}$ of the partition $\hat{\mathcal{C}}$ is called a culture. We impose the additional requirement to the partition $\hat{\mathcal{C}}$ that all labels within a culture $\hat{c}$ are different, so that cultures can be represented as sets of labels. Labels that belong to the same class of $\hat{\mathcal{C}}$ are said to represent the same culture, and the partition $\hat{\mathcal{C}}$ thus represents the syntactic equivalence of the microbial labelling system. The set of all cultures $\hat{\mathcal{C}}$ is further partitioned into classes that harbour the cultures that originate from the same isolate in pure culture. This partition is denoted as $\hat{\mathcal{S}}$, and we have that

$$\hat{\mathcal{S}} \;=\; \{\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_{|\hat{\mathcal{S}}|}\}\,, \tag{2.4}$$

$$\text{where } \begin{cases} \hat{s}_i \subset \hat{\mathcal{L}} & 1 \leq i \leq |\hat{\mathcal{S}}| \\ \hat{s}_i \cap \hat{s}_j = \emptyset & 1 \leq i, j \leq |\hat{\mathcal{S}}|,\ i \neq j \\ (\forall \hat{c} \in \hat{\mathcal{C}})(\exists \hat{s} \in \hat{\mathcal{S}})(\hat{c} \subseteq \hat{s}) \\ \bigcup_{i=1}^{|\hat{\mathcal{S}}|} \hat{s}_i = \hat{\mathcal{L}}\,. \end{cases}$$

Each class of the partition $\hat{\mathcal{S}}$ is called a strain, and in analogy with the cultures we require that all labels in a strain class are different, so that strains equally can be represented as sets of labels. Labels that belong to the same class of $\hat{\mathcal{S}}$ are said to represent the same strain, hence the partition $\hat{\mathcal{S}}$ represents the synonym equivalence of the labelling system. As such, the partitions $\hat{\mathcal{C}}$ and $\hat{\mathcal{S}}$ impose a hierarchical partition upon the label space $\hat{\mathcal{L}}$, where $\hat{\mathcal{C}}$ is the subpartition of $\hat{\mathcal{S}}$. Accordingly, the problem dealt with in the framework of this chapter can be described as the process of incrementally learning the triplet ($\mathcal{L}$,$\mathcal{C}$,$\mathcal{S}$) from the partial information supplied by a number of autonomous and heterogeneous data sources. Herein, the triplet ($\mathcal{L}$,$\mathcal{C}$,$\mathcal{S}$) is a representation of the partial knowledge we have learned so far about the triplet ($\hat{\mathcal{L}}$,$\hat{\mathcal{C}}$,$\hat{\mathcal{S}}$) as it is observed in the real-world.

Before we come to the detailed description of an algorithm that automates the incremental accumulation of label equivalence knowledge extracted from different data sources into the triplet ($\mathcal{L}$,$\mathcal{C}$,$\mathcal{S}$), we first set off with a short introduction on the representation of

this triplet within the integrated strain database. For reasons of compactness, we will denote the status of the integrated strain database as $\mathcal{I}$ in the rest of this chapter. Initially, before we have processed any information about the entities and their equivalences in the real-world, the conceptual knowledge represented in $\mathcal{I}$ is empty, meaning that ($\mathcal{L}$,$\mathcal{C}$,$\mathcal{S}$) $\equiv$ ($\emptyset$,$\emptyset$,$\emptyset$). The Object Model formulated by the Object Data Management Group (ODMG; [11]) discriminates literals from classes by the fact that each instance of a class has its own identity, which is not the case for literal instances. Based on this model, the process of archiving newly learned concepts as instances of first class objects was coined *capacity augmentation* by Josifovski and Risch [42]. In the specific case of learning the equivalence classes $\mathcal{C}$ and $\mathcal{S}$ over the label space $\mathcal{L}$, this can simply be achieved by the annotation of each learned label l with an object identifier $\mathcal{C}_l$ for the cultures and an object identifier $\mathcal{S}_l$ for the strains. We call $\mathcal{C}_l$ the culture identifier (CID) of the label l, and $\mathcal{S}_l$ the strain identifier (SID) of the label l. If two labels share the same culture identifier in the integrated strain database $\mathcal{I}$, they are considered to represent the same culture, while if two labels share the same strain identifier they are considered to be synonymous labels of the same strain. Remark that the hierarchical clustering of strains and cultures dictates that if two labels share the same culture identifier, they necessarily also have to share the same strain identifier. The application of object identifiers has the main advantage that data which is relevant for the integrated strain database can be associated locally to strains and cultures. Capacity augmentation also enables the use of culture and strain objects within attributes and methods that are locally stored within $\mathcal{I}$, by treating them just as ordinary object identifiers. In addition, if strain and culture identifiers are made publicly available for reuse outside the scope of the integrated strain database, they may contribute to the implementation of an integration schema that helps resolve the current application of homonymous and synonymous labels for encoding strain and culture information within peripheral databases of microbial information, such as public sequence databases [40] and scientific literature databases, an issue that is scrutinized in section 2.5. This opens a door for the execution of advanced queries that abridge the boundaries of distributed and heterogeneous microbial databases, which is a preliminary condition for the development of intelligent data mining applications [16].

We now describe an algorithm designed for the incremental update of the entities and equivalences stored in the integrated strain database $\mathcal{I}$, as a simulation of the learning process of possible new information extracted from a series of autonomous and heterogeneous data sources. First we define a *record* as any subset of a strain class in $\hat{\mathcal{S}}$, and we denote the collection of all records as $\mathcal{R}$ with the following notation

$$\mathcal{R} \;=\; \{r_1, r_2, \ldots, r_{|\mathcal{R}|}\}\,, \tag{2.5}$$

$$\text{where } (\forall r \in \mathcal{R})(\exists \hat{s} \in \hat{\mathcal{S}})(r \subseteq \hat{s})\,.$$

As such, a record is a set of labels which by definition belong to the same strain, and the collection $\mathcal{R}$ represents all the accumulated information we have processed to learn the knowledge stored in the integrated strain database $\mathcal{I}$. A record that only contains ambiguous labels is called an *ambiguous record*. Remark also that for records we do not require that $r_i \cap r_j = \emptyset$ $(1 \leq i, j \leq |\mathcal{R}|)$, if $i \neq j$, which means that the synonym equivalence information supplied by different records may be overlapping. Moreover, if $r \subset \hat{s}$ for a given $r \in \mathcal{R}$ and $\hat{s} \in \hat{\mathcal{S}}$, we say that the synonym equivalence information supplied by the

record r is incomplete. Finally, we define the set of all *data sources* $\mathcal{D}$ as a partition of $\mathcal{R}$, so that

$$\mathcal{D} \;=\; \{d_1, d_2, \ldots, d_{|\mathcal{D}|}\}, \tag{2.6}$$

$$\text{where} \left\{ \begin{array}{ll} d_i \subset \mathcal{R}, & 1 \leq i \leq |\mathcal{D}| \\ d_i \cap d_j = \emptyset, & 1 \leq i, j \leq |\mathcal{D}|,\ i \neq j \\ \bigcup_{i=1}^{|\mathcal{D}|} d_i = \mathcal{R}\ . & \end{array} \right.$$

The pseudo-code procedure `IncrementEquivalence` gives a detailed description of an algorithm for updating the prior knowledge recorded in the integrated strain database $\mathcal{I}$, on the basis of the new equivalence information learned from a record $r = \{l_1, l_2, \ldots, l_{|r|}\}$. As was described previously, all labels in the record r should be regarded as being synonym labels of the same strain. On exit of the procedure `IncrementEquivalence`, all labels of the record r will be incorporated within the integrated strain database $\mathcal{I}$ and annotated with the same strain identifier $\mathcal{S}_r$, which we call the strain identifier of the record r. Additionally, the culture identifier for each of the labels in the record r will be available on exit of the procedure.

```
1  IncrementEquivalence(r, I)
2    Input: r = {l_1, l_2, ..., l_|r|}  : record of synonym labels
3           I                           : current status of (L,C,S)
4    Output on exit: s  : S_r
5                    c_i : C_{l_i}      1 ≤ i ≤ |r|
6                    I : updated status of (L,C,S)
7
8     s := NULL;
9     unique_found := false;
10    for i from 1 to |r| do
11      c_i := NULL;
12
13    for i from 1 to |r| do
14      if is_unique(l_i) then
15        unique_found := true;
16        c_i := culture.find(l_i);
17        if c_i ≠ NULL then
18          s_temp := strain.find(c_i);
19          if s = NULL then s := s_temp;
20          else if s ≠ s_temp then s := strain.union(s, s_temp);
21
22    if not(unique_found) then Exit;
23    if s = NULL then s := strain.create_new;
24
25    for i from 1 to |r| do
26      if not(is_unique(l_i)) then
27        c_i := culture.find(l_i, s);
28      if c_i = NULL then
29        c_i := culture.create_new(l_i, s);
```

On lines 8–11, the algorithm starts with the initialisation of the object identifier variables s and c_i ($1 \leq i \leq |R|$) to the value `NULL`. This value indicates that no object is referenced by the corresponding object identifier variable. The boolean flag `unique_found` is initially set to false. During the execution of the procedure, the variable $s_{\texttt{temp}}$ is used for temporary storage of a local strain object identifier.

The continuation of the algorithm sequentially loops twice through the set of labels in the record r, where the first loop (lines 13–23) only works upon the unique labels within the record, as to avoid semantically incorrect strain merges caused by homonymous labels. In order to discriminate the unique from the ambiguous labels, the function `is_unique` returns a boolean value that determines whether the argument label l is unique (return value true) or ambiguous (return value false). Theoretically, it is impossible to know in advance whether or not a given label will be unique, as we have no complete prior knowledge about the label space $\hat{\mathcal{L}}$. However, the implementation of the function `is_unique` in the integrated strain database simply considers all labels that have an acronym which occurs within a predefined list as unique. All other labels are conservatively regarded as being ambiguous. The list of acronyms which define unique labels, was originally fed with acronyms of the instances that are mentioned in the directory of culture collections published by the World Data Centre for Microorganisms (WDCM; [74]). Additionally, the option to manually add or remove acronyms to or from this list was built into the integrated strain database. Appendix C contains a snapshot of the list of acronyms used for prediction of label uniqueness in the integrated strain database. This approximative and updateable way of estimating the uniqueness of labels includes some hazards, but has proven to be workable in practice. Following, for each unique label $l \in r$, the associated culture identifier $\mathcal{C}_l$ and strain identifier $\mathcal{S}_l$ are looked up within the integrated strain database $\mathcal{I}$. This is accomplished by an initial transformation of the label into its normalized syntactic form, as discussed in section 2.2.1, followed by a search for the presence of the normalized label in $\mathcal{L}$. In the pseudo-code, we have split the lookup of object identifiers of a given label over two separate functions. The function `culture.find`(l) returns the culture identifier of the unique label l, if the normalized form of the label already occurs in $\mathcal{L}$. Otherwise, the value `NULL` is returned. The function `strain.find`(c) returns the strain identifier of the culture with culture identifier c. This makes sense given the hierarchical relationship between strains and cultures. In case the argument passed to the function `strain.find` is `NULL`, the `NULL` value is returned. The two functions `strain.find` and `culture.find` can easily be composed in order to directly lookup the strain identifier of a given label l, in the following way

$$\texttt{strain.find}(l) \equiv \texttt{strain.find}(\texttt{culture.find}(l))\,. \tag{2.7}$$

In case the strain identifier $s_{\texttt{temp}}$ found for the label l_i of the record r differs from the strain identifier s found for the previously processed labels, we have discovered a situation in the integrated strain database that contradicts with the definition of a record, namely that all labels within the same record should represent the same strain. This can happen in practical situations where the synonym information provided by the records is incomplete, which may result in that fact that the necessary evidence for synonymy of the labels of the strains with identifiers s and $s_{\texttt{temp}}$ was not yet discovered from processing previous record information with the procedure `IncrementEquivalence`. The algorithm resolves such conflicting situations by merging the corresponding strain classes within the

integrated strain database. This unification of strain classes is executed by the function call `strain.union`($s, s_{\texttt{temp}}$) on line 20. Though very efficient algorithms exist for the calculation of the transitive closure in cases where all pairwise equalities are known in beforehand [19, 24, 48, 51], the chosen names of the functions in the pseudo-code of the procedure `IncrementEquivalence` already suggest that we have opted for the implementation of a union-find data structure [14, 38] for maintaining the transitivity of the strain equivalence relation in the integrated strain database. After all, these union-find data structures allow for the incremental management of an equivalence relation, which is more appropriate than recalculating the whole transitive closure in environments where new or updated information regularly shows up. Note that the problem of incrementally computing the connected components of a graph is harder than just finding the connected components [57]. Moreover, the merge of two strain classes might also trigger the merge of some of their enclosed culture classes, as we have required that (normalized) labels must be unique within the scope of a strain class. Although somewhat hidden in the pseudo-code shown here, the maintainance of the culture equivalence relation was thus also implemented using another union-find data structure in the integrated strain database. If no strain identifier was found after an initial processing of the unique labels, it means that currently no strain is known in $\mathcal{I}$ that has a unique synonym label in common with the record passed to the procedure `IncrementEquivalence`. In other words $\mathcal{L} \cap U(r) = \emptyset$. For such cases, the function `strain.create_new` establishes that a new strain object is created in $\mathcal{I}$ as an empty subset of $\mathcal{L}$. The new strain identifier is returned as the output value of the procedure `strain.create_new`, and will later on during the execution of the procedure be applied to fill the newly created strain class with the normalized labels of r.

After a first run through the unique labels of the record r has determined the corresponding object identifier $\mathcal{S}_r$ of the strain within the integrated strain database $\mathcal{I}$, with the potential side-effect of creating a new strain object with an empty associated label set, the resulting strain can be regarded as the necessary context for looking up the remaining ambiguous labels of the record r. After all, we have required that normalized labels are different within the classes of the partition $\mathcal{S}$. The overloaded version of the function `culture.find`(l, s) therefore looks up the culture identifier associated with the label l, by searching for an occurrence of the normalized form of the label within the limited scope of $\mathcal{L}$ that corresponds with the subset of labels associated to the strain identifier s. Again, the `NULL` value is returned if the search did not result in a matching normalized label in $\mathcal{I}$. Finally on line 29, the function `culture.create_new`(l, s) is called for each label l for which no corresponding strain and culture objects were found during previous search operations in $\mathcal{L}$ performed while executing the procedure `IncrementEquivalence`. This function adds the normalized form of the label l to the set of all learned labels $\mathcal{L}$, and tags it with the strain identifier s and the culture identifier of a newly created culture object. In this way, each label of the record r is guaranteed to be represented by a culture in $\mathcal{C}$, whereas all labels of the record belong to the same class of $\mathcal{S}$ associated with the strain identifier stored in the local variable s that is returned by the procedure.

If an ambiguous record is passed as an argument to the algorithm, this would result in the creation of a strain class containing solely ambiguous labels in the integrated strain database $\mathcal{I}$. Worse, if the same ambiguous record is repeatedly passed as an argument to

the function `IncrementEquivalence` in the frame of the iterative processing of the data source wherein this record is embedded, new strain objects with the same information content would repeatedly be created in $\mathcal{I}$. Such strain objects are ambiguous themselves, as the object identifiers associated to these strains or to the cultures within them would never be retrieved by the functions `culture.find` and `strain.find` during execution of the procedure `IncrementEquivalence`. This can be considered as a worse situation than not learning anything from an ambiguous record. Therefore, we have built in the option to stop the execution of the procedure on line 22, in case the `unique_found` flag indicates that the argument record r is ambiguous.

We can further improve the performance of the algorithm, by adopting a priority searching strategy for the retrieval of the culture identifier associated to a given label. Monge *et al.* [57] came up with the idea that an extra evaluation of the pairwise equivalence of two labels can be avoided, if that equivalence already follows from the transitivity of the currently found equivalence relation. A similar pruning of the search space can in case of the procedure `IncrementEquivalence` be accomplished by replacing line 16 of the algorithm with the following piece of pseudo-code.

if $s \neq$ `NULL` **then**
 c_i := `culture.find`(l_i, s);
 if c_i = `NULL` **then** c_i := `culture.find`(l_i);
else c_i := `culture.find`(l_i);

The above code excerpt states that in case a strain identifier for the record r was previously found in the loop wherein this search is nested ($s \neq$ `NULL`), we can pre-emptively perform a restricted search on the subset of $\mathcal{L}$ associated with the strain identifier s. This low-cost limited search is then only followed by a more time consuming full scan of $\mathcal{L}$, in case no corresponding label was detected in the pruned search space. This idea is supported by the prerequisite that all labels of the record r have to belong to the same class in $\mathcal{S}$ after execution of the procedure `IncrementEquivalence`, so that we can assume that there is a realistic chance that some or all of the labels of r will already belong to the same class of $\mathcal{S}$ in $\mathcal{I}$ due to the records that were previously processed by the algorithm. Although the extra limited search will result in a minor overhead when the amount of knowledge learned in $\mathcal{I}$ is yet minimal so that only few overlapping information is processed by calls to the procedure `IncrementEquivalence`, this is by far outweighed by the significant savings in the number of times that a full scan of $\mathcal{L}$ must be applied, in the case where $\mathcal{I}$ gets saturated and the majority of the processed information is already captured within the integrated strain database. Also remark that this extra restricted search by no means impairs the accuracy of the algorithm.

2.3 Error detection/correction strategies

In the previous section we have explained in detail how a complete equational theory for the labelling system used in the field of microbiology can be jointly built up from the integration of partial and overlapping information extracted from a series of autonomous and

heterogeneous data sources. However, from the first draft versions of the integrated strain database, it became immediately evident that quite a large number of inconsistencies were showing up in the information retrieved from the originating data sources. This observation highly compromised the quality of the resulting database, and undermined our initial goal of integrating the fragmented knowledge into a complete and correct information service provider. Hernandez and Stolfo [37] state that in real-world datasets, one obviously cannot estimate the true equivalence classes with high precision without a time consuming and expensive human inspection and validation process. With this in mind, we will devote this section to a discussion on some of the vulnerabilities of the integration process and unravel some strategies that we have additionally deployed during the construction of the integrated strain database, in order to cope with information attained from data providers that are not completely trustworthy.

2.3.1 Basic error detection and correction

Under ideal circumstances, the entities and their equivalences learned in the integrated strain database $\mathcal{I}$, which we have represented by means of the triplet ($\mathcal{L}$,$\mathcal{C}$,$\mathcal{S}$), should be a complete and correct reflection of their counterparts ($\hat{\mathcal{L}}$,$\hat{\mathcal{C}}$,$\hat{\mathcal{S}}$) in the real world. However, in practice we might encounter strain classes $s \in \mathcal{S}$ which only contain a subset of all the labels assigned to their corresponding real-world strain $\hat{s} \in \hat{\mathcal{S}}$. In such cases, we say that the known synonym information of the strain $s \in \mathcal{S}$ is *incomplete*. Information collected based on incomplete knowledge about the synonym labels assigned to a strain, might be incomplete as well. The most obvious way to accomodate for incomplete information in the integrated strain database, is to continuously monitor whether new data sources are made publicly available or the information content of previously processed data sources has been updated. Synonym information extracted from the new or updated records can accordingly be incorporated in the integrated strain database by application of the procedure `IncrementEquivalence`. One particular situation related to the problem of incomplete strains occurs when the integrated strain database contains two different strain classes $s_1, s_2 \in \mathcal{S}$ which are both subsets of the same real-world strain $\hat{s} \in \hat{\mathcal{S}}$. We say that such equivalence classes s_1 and s_2 are *false negatives*, because the integrated strain database has failed to recognize that both classes deal with the same real-world entity. False negative equivalence classes may result from the fact that the records that support those classes do not share enough common unique labels. Otherwise, the equivalence classes would have been merged due to the calculation of the transitive closure during the execution of the procedure `IncrementEquivalence`. As an example, the observation that the equivalence classes shown in Table 2.3 share the ambiguous label `HD-1` (which is not taken into account for merging strain classes in the `IncrementEquivalence` procedure), together with the additional knowledge that their corresponding strains are identified to belong to the same species, might provide sufficient evidence to a specialist in the domain to conclude that in this case we are dealing with false negative strain classes. To resolve the detection of this kind of false negatives, the necessary tools have been implemented in the integrated strain database to support manual merges of equivalence classes. Remark however that there is also some chance that the false negative equivalences of the example will be auto-

SID	species name	synonym labels
20126	*Bacillus thuringiensis*	`ATCC 39756, CMCC 1615,` **HD-1**
41574	*Bacillus thuringiensis*	`ATCC 33679, CCRC 14616, CECT 4454, Dulmadge` **HD-1**, **HD-1** `KCTC 1507, NRRL B-3792, NCAIM B.01262, MKBT B-0044`
62385	*Bacillus thuringiensis*	`DSM 6102,` **HD-1**

Table 2.3: Example of the occurrence of false negative strain classes in the integrated strain database.

matically resolved, in case the missing unique synonymy evidence is provided by a record that is processed at some future point in time.

On the contrary, due to typographical mistakes or transcription errors the equational theory might decide that two labels which belong to the same record are synonyms, even though they may not represent the same strain in the real-world. As a result, some strain classes in the integrated strain database might be composed of two or more true equivalence classes, whereby the labels of several true strains have illegitimately been merged into the same equivalence class caused by the calculation of the transitive closure. A strain class $s \in \mathcal{S}$ is said to contain *false positive* equivalences, if at least two different strains $\hat{s}_1, \hat{s}_2 \in \hat{\mathcal{S}}$ exist in the real-world wherefore $s \cap \hat{s}_1 \neq \emptyset$ and $s \cap \hat{s}_2 \neq \emptyset$. In the rest of this subsection we will focus on a strategy we have developed for the detection and correction of this kind of false positive strain classes in the integrated strain database. After all, the properties of a strain that are collected using incorrectly merged synonym information, might be incorrectly joined together as well.

A systematic way to check the semantic accuracy when several data sources are available, is to compare and verify the information related to the same entity provided by the different resources [53]. In order to explain how this idea was implemented in the integrated strain database, we first need to introduce the concept and construction of synonym cross-reference matrices. For a given strain class $s \in \mathcal{S}$, we define $\mathcal{R}_s$ as the subset of records in $\mathcal{R}$ that contains all evidence that was found for the synonym labels of the strain s, so that we have that

$$\mathcal{R}_s = \{r \in \mathcal{R} \,|\, \mathcal{S}_r = s\} \equiv \{r_1, r_2, \ldots, r_m\}\,, \tag{2.8}$$

where we have introduced the short notation $m \equiv |\mathcal{R}_s|$ for the number of records that are contributing evidence for the synonym labels of the strain s. As an example, Table 2.1 lists all the records we have found with evidence for the synonym labels of the *Bacillus cereus* type strain. Generally, the synonym label evidence for a strain s thus exists of m sets of labels

$$\left\{\begin{array}{rcl} r_1 & = & \{l_1^1, l_2^1, \ldots, l_{|r_1|}^1\} \\ r_2 & = & \{l_1^2, l_2^2, \ldots, l_{|r_2|}^2\} \\ & \vdots & \\ r_m & = & \{l_1^m, l_2^m, \ldots, l_{|r_m|}^m\}\,, \end{array}\right. \tag{2.9}$$

which are integrated by means of the `IncrementEquivalence` procedure into a set of cultures $\mathcal{C}_s$, which collectively constitute the strain s. The set $\mathcal{C}_s$ can thus be constructed as

$$\mathcal{C}_s = \bigcup_{i=1}^{m} \bigcup_{j=1}^{|r_i|} \texttt{culture.find(}l_j^i\texttt{,}\, s\texttt{)} \equiv \{c_1, c_2, \ldots, c_n\}\,, \tag{2.10}$$

where we have introduced another short notation, $n \equiv |\mathcal{C}_s|$, for the number of culture classes that form the subdivisions of the strain s. Accordingly, we can define the *synonym cross-reference matrix* of the strain s as the $(m \times n)$ binary matrix $(b_{ij})_{i=1,\ldots,m}^{j=1,\ldots,n}$, with elements defined in the following way

$$b_{ij} = \begin{cases} 0 & \Leftrightarrow \texttt{culture.find(}l_k^i\texttt{,}s\texttt{)} \neq c_j\,, \text{ for all } 1 \leq k \leq |r_i| \\ 1 & \texttt{otherwise}\,. \end{cases} \tag{2.11}$$

As such, the boolean value b_{ij} expresses whether or not the record r_i contains a label that is mapped to the culture c_j in the integrated strain database. Figure 2.1 shows the graphical representation of an example synonym cross-reference matrix for the *Bacillus cereus* type strain. Hereby, we have annotated the rows of the synonym cross-reference matrix that are extracted from an online catalogue of a culture collection with the label assigned to the catalogue record (and thus to the culture stored and distributed by that culture collection), together with the identification of the strain as mentioned in the catalogue. Rows that correspond with records taken from the Bacterial Nomenclature Up-to-date [73] data source are marked with the descriptor `taxa (DSMZ)`, followed by the name of the taxon as indicated in the data source record. It should be noticed that for this case, all instances have uniformly identified the strain as *Bacillus cereus*. This completely follows the expectations, because the example is dealing with a type strain, which is by definition the name-bearer of the species it belongs to. Each column of a synonym cross-reference matrix corresponds with a culture of $\mathcal{C}_s$, and is consequently tagged in Figure 2.1 with the most representative label for the culture as determined within the integrated strain database. In the graphical representation of the synonym cross-reference matrices, we have indicated 1-valued matrix elements b_{ij} with colored boxes, while positions of the matrix elements that equal to zero are left blanc. Self-references, i.e. 1-valued matrix elements for which the row label of the catalogue record is also a label of the culture of the corresponding column of the matrix, are represented by means of a green box, whereas the other cross-references are marked with a black box.

In this respect, synonym cross-reference matrices describe the presence or absence of the pairwise cross-references that constitute the equational theory of synonym labels in an easy-to-interpret manner. One can easily derive from a glancing inspection of the cross-reference tables that the list of synonym strain labels is far from complete in most data sources, which explains the urge to calculate the transitive closure of the partial equivalences gathered from different data sources. This presence/absence can even be quantified for a given strain s using the synonym cross-reference matrices in the following way. For a given record r_i of the strain, the value $\frac{1}{m}\sum_{j=1}^{m} b_{ij}$ summed over all columns of the corresponding row in the cross-reference matrix, expresses the completeness of the synonyms mentioned in the record, in comparison with all known synonym labels of the strain in the integrated strain database. Alternatively, the dual value $\frac{1}{n}\sum_{i=1}^{n} b_{ij}$ summed over all rows of the j-th column in the cross-reference matrix, gives an estimation of the presence of any label of the corresponding culture c_j as a synonym in the inspected data sources. These values have been calculated for all rows and columns in the graphical representation of the synonym cross-reference matrix of the *Bacillus cereus* type strain. From this example we can easily find out that the label `ATCC 14579`$^{\mathsf{T}}$ has been referenced by all inspected records of the *Bacillus cereus* type strain, whereas the `ATCC` culture collection (see Table

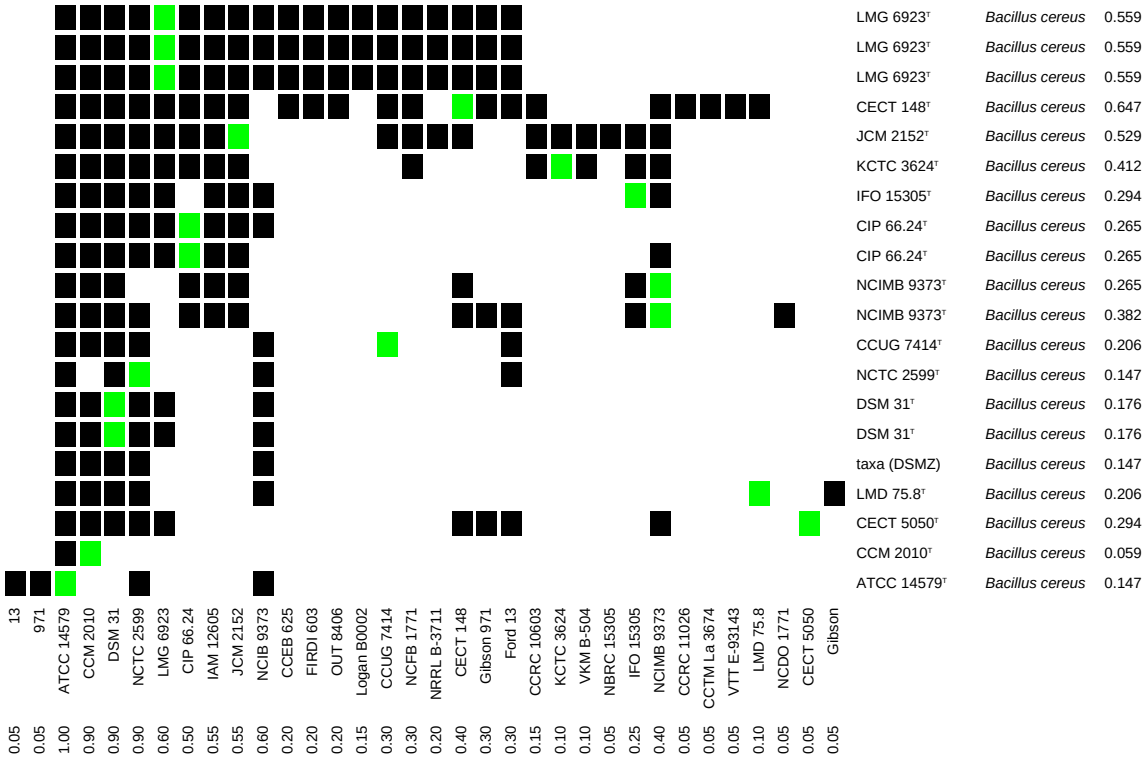


Figure 2.1: Synonym cross-reference matrix for the *Bacillus cereus* type strain.

2.6) record itself only mentions 14.7% of the synonyms known by the integrated strain database. On the other hand, the `CECT` culture collection (see Table 2.6) has managed to include 64.7% of all known labels assigned to the *Bacillus cereus* type strain in the catalogue entry of its proper culture `CECT 148`$^{\mathsf{T}}$ (which is the best result of all *Bacillus cereus* type strain records processed), while the `CECT 148`$^{\mathsf{T}}$ label itself only occurs in 40% of the *Bacillus cereus* type strain records of the scanned data sources. Also note the difference in the synonym information content provided by the records labelled `CECT 148`$^{\mathsf{T}}$ and `CECT 5050`$^{\mathsf{T}}$, notwithstanding the fact that both records represent different cultures of the same strain distributed by the same culture collection.

The completeness of the evidence found for the synonym labels of a given strain s within all records of the inspected data sources, can be quantified as

$$\omega_s = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} b_{ij} \,. \tag{2.12}$$

From the synonym cross-reference matrix of the *Bacillus cereus* type strain it becomes however immediatly evident that this value may show a slight underestimation of the completeness, due to the fact that the equational theory has for example missed the syntactical equivalence of the labels `971`, `Gibson` and `Gibson 971` as being labels assigned to the same culture. This situation occurs more often for ambiguous labels than for unique labels, so that a more reliable estimation of the completeness of synonym evidence can be made if

we restrict the calculation of ω_s to the unique labels

$$\omega_s^u = \frac{1}{|\mathcal{R}_s|\,|U(s)|} \sum_{r \in \mathcal{R}_s} |U(r)|\,, \tag{2.13}$$

where the superscript ^{u} indicates the limitation of the scope to the unique labels. $U(s)$ denotes the subset of unique labels of the strain s in formula (2.13), while $U(r)$ denotes the subset of unique labels of the record r. For the *Bacillus cereus* type strain example we find that $\omega_s^u = \frac{192}{20\times26} \approx 0.37$, which means that for this case only 37% of the pairwise synonym equivalences that are generated by the calculation of the transitive closure are directly retrieved from the inspected data sources. In section 2.4 we will demonstrate that the synonym completeness is generally low to moderate for all strains. Strain classes for which the completeness w_s^u falls below an emperically determined threshold, could be suspected of containing inconsistencies. After all, when strain classes are illegitimately merged in the integrated strain database due to erroneous synonym information in the data sources, one may expect that the total amount of correctly paired synonyms retrieved from the data sources far outweighs the number of incorrect pairs. Hence, the transitive closure will result in a steep drop of the amount of evidence found for the merged equivalence classes, with respect to the evidence found for the correctly splitted equivalence classes.

With the quantifier w_s^u for the completeness of the synonym evidence found for a strain class, we have discovered a first indicator that can be deployed for the detection of internal inconsistencies within the strain classes of the integrated strain database. Table 2.4 depicts and excerpt of the records in $\mathcal{R}_s$ of an example strain class in the integrated strain database, with an evidence completeness $w_s^u = \frac{454}{45\times96} \approx 0.1051$. The completeness value for this strain class is extremely low, so that we might assume here that some of the underlying data sources could be disseminating erroneous synonym information, which has possibly driven the integration process into the creation of many more false positive equivalences within the strain class. From the contextual information we have included into Table 2.4, it is indeed not difficult to derive that the `IncrementEquivalence` procedure has probably illegitimately merged the synonym labels of three different *Streptomyces* spp. type strains into a single equivalence class. With this in mind, strain classes with mixed taxonomic identifications in their corresponding data sources have also been regarded as potential intruders of the equivalence relation of label synonyms. This is a good example of how meta-data might be applied for spotting errors within the integrated strain database. Note however that for a given strain class, variation in the taxonomic naming between data sources does not neccesarily reflect the presence of errors in the integrated strain database, but might simply be a consequence of differences in taxonomic opinion, such as the use of synonym names for the same taxon or the application of alternative methodologies and insights for the identification of microorganisms. Where the completeness qualifier w_s^u might fail to detect inconsistencies within the union of strain classes with different sizes, and the detection of mixed species classes is vulnerable for different opinions reflected in the contextual information and is unable to point out the merge of strains that belong to the same species (which we know from our experience is not all that rare), an alternative indicator that has been succesfully implemented for the detection of inconsistencies in strain classes of the integrated strain database, searches for non-overlapping records of the same data source that are linked to the same strain class. For these records, the evidence for

merging the strain classes has not been directly found within the data source at hand. The strain class shown in Table 2.4 abounds of non-overlapping records. For example none of the records corresponding with the catalogue entries ATCC 3320^{T}, ATCC 10745^{T} or ATCC 19762^{T} has any label in common. Note that also in this case the presence of non-overlapping records from the same data source related to the same strain class, should not automatically reflect errors in strain class itself, but might be due to the shortcoming of a data source to include the expected internal synonym references between its own records.

Linkage of the integrated strain database with external data sources that provide additional strain information (called *peripheral databases* for short in the context of this chapter), does not only allow a better management of the peripheral data sources and leads to the direct advantage of enabling more advanced and accurate queries, but also opens up additional ways of performing error detection in the integrated strain database. In section 2.5 we will demonstrate how the settlement of uniform cross-references may enable advanced queries that bridge over the borders of both the integrated strain database that is discussed in the context of this chapter and the public International Nucleotide Sequence Database [40]. As a result, one can easily collect all 16S rRNA sequences for a given strain and calculate the homogeneity amongst these sequences, which should presumably be highly similar. Strain classes for which the compared sequences turn out to be too heterogeneous could accordingly be checked on the correctness of their synonym labels in the integrated strain database. There could however be a number of other reasons that cause the heterogeneity amongst the sequences linked to the same strain class, such as poor quality of some of the sequences or mistakes made in the cross-references between the integrated strain database and the sequence database, e.g. due to typographic errors in the strain label mentioned in the sequence database. Detection of these other kind of inconsistencies could have value in its own right, but is left out of the scope of this chapter.

Having summed up a series of indicators for the discovery of possible inconsistencies in the equivalence classes of the integrated strain database, it is still not always trivial to confirm whether or not a mistake was made during the construction of some strain classes, or to point out where the exact mistakes were made in the data sources, which caused the illegitimate merge of strain classes such as the one shown in Table 2.4. This latter information is necessary in order to enable the correct split of erroneously merged strain classes in their true composing strain classes. In the integrated strain database we have therefore implemented an error detection/correction strategy that builds on top of the synonym cross-references matrices that were discussed before. After construction of the cross-reference matrix of a strain class that is suspected of containing false positive equivalence information, the error detection/correction strategy uses the transversal grouping (or two-way joining) technique [35] for the simultaneous partitioning of both the binary row and column vectors of the matrix. In this way, data source records are clustered according to their synonym information content, whereas at the same time the different cultures that compose the strain class are grouped according to their occurrences in the inspected data sources. Alternative partitioning methods for the classification of binary vectors can be embedded as options of the error detection/correction strategy, such as hierarchical clustering algorithms [3, 67] in combination with manual or automated selection of an optimal α-cut [22, 33], or non-hierarchical methods such as k-means clustering [3, 35] or classification

record ID	species name	synonym labels
ATCC 3320[T]	*Streptomyces flavovirens*	IMRU 3320; ATCC 19758; CBS 129.20; CBS 496.68; IFO 12771; ISP 5062; PSA 217; RIA 1038
ATCC 10745[T]	*Streptomyces fradiae*	CBS 498.68; ETH 13472; IFO 3718; IFO 12773; IMI 61202; KCC S-0133; NCIB 8233; NRRL B-1195; PSA 61; PSA 156; RIA 1040
ATCC 19760[T]	*Streptomyces fradiae*	ISP 5063
ATCC 19762[T]	*Streptomyces griseobrunneus*	ISP 5066; CBS 500.68; ETH 31437; IFO 12775; IMRU 3068; KCC S-0380; RIA 1042
CBS 129.20[T]	*Streptomyces flavovirens*	CBS 189.75; CBS 496.68; CBS 279.30; ISP 5062; ATCC 3320; ATCC 19758; IMRU 3320; IFO 3716; IFO 3197; IFO 3412; IFO 12771; ETH 10248; ETH 24134; ETH 31593; DSM 40062; RIA 635; RIA 1038
CBS 189.75[T]	*Streptomyces flavovirens*	ATCC 3320; ATCC 19758; ISP 5062; IMRU 3320; CBS 496.68; CBS 279.30; CBS 129.20; IFO 3197; IFO 3412; IFO 3716; IFO 12771; ETH 10248; ETH 24134; ETH 31593; DSM 40062; RIA 635; RIA 1038
CBS 498.68[T]	*Streptomyces fradiae*	ISP 5063; IMI 61202; ATCC 10745; ATCC 19760; IFO 3439; IFO 3718; IFO 12773; IMRU 3535; NRRL B-1195; ETH 13363; ETH 13472; DSM 40063; RIA 97; RIA 1040; CBS 414.54
CBS 500.68[T]	*Streptomyces griseobrunneus*	ISP 5066; ATCC 19762; IFO 12775; ETH 31581; IMRU 3068; DSM 40066; RIA 1042; KCC S-0380
CCM 3174[T]	*Streptomyces fradiae*	ATCC 10745
CCM 3243[T]	*Streptomyces flavovirens*	ATCC 3320
CCUG 11105[T]	*Streptomyces griseobrunneus*	ISP 5066; HJ Kutzner; ATCC 19762; DSM 40066; CBS 500.68; ETH 31437; IFO 12775
CECT 3197[T]	*Streptomyces fradiae*	ATCC 10745; ATCC 19760; Boots FD276; CBS 414.54; CBS 498.68; CCM 3174; CCRC 12196; CCTM La 2925; DSM 40063; ETH 13363; ETH 13472; HMGB B923; HUT 6095; IAM 0083; IFO 3439; IFO 3718; IFO 12773; IMET 42051; IMI 61202; IMRU 3535; ISP 5063; JCM 4133; JCM 4579; KCC S-0133; KCC S-0579;
DSM 40062[T]	*Streptomyces flavovirens*	ATCC 19758; ATCC 3320; CBS 129.20; CBS 496.68; IFO 12771; IFO 3412; IMRU 3320; ISP 5062; KCC S-0035; RIA 1038
DSM 40063[T]	*Streptomyces fradiae*	ATCC 10745; ATCC 19760; CBS 498.68; IFO 12773; IMRU 3535; ISP 5063; JCM 4133; JCM 4579; NCIB 8233; NRRL B-1195; RIA 1040; ETH 13363; ETH 13472; ETH 28510
DSM 40066[T]	*Streptomyces griseobrunneus*	ATCC 19762; CBS 500.68; CBS 500.68; IFO 12775; IMRU 3068; ISP 5066; JCM 4380; RIA 1042; ETH 31437; ETH 31581
DSM 46372[T]	*Streptomyces fradiae*	CCM 3174; HMGB B 922; IFO 3439; IMET 40283; LBG A 3013; NCIB 8233; NRRL B-1195; ETH 13363
IFO 3197[T]	*Streptomyces flavovirens*	
IFO 3412[T]	*Streptomyces flavovirens*	ATCC 3320; IAM W5-7
IFO 3439[T]	*Streptomyces fradiae*	NRRL B-1195
IFO 3716[T]	*Streptomyces flavovirens*	ATCC 3320
IFO 12174[T]	*Streptomyces fradiae*	OUT 8322; RIA 97
IFO 12771[T]	*Streptomyces flavovirens*	ATCC 19758; ATCC 3320; CBS 129.20; CBS 496.68; RIA 1038
IFO 12773[T]	*Streptomyces fradiae*	ATCC 10745; ATCC 19760; CBS 498.68; IFO 3718; RIA 1040
IFO 12775[T]	*Streptomyces griseobrunneus*	ATCC 19762; CBS 500.68; RIA 1042
IMI 061202[T]	*Streptomyces fradiae*	ATCC 10745
JCM 4035[T]	*Streptomyces flavovirens*	AS 4.575; ATCC 19758; ATCC 3320; CBS 129.20; CBS 279.30; CBS 496.68; CCM 3243; CCRC 13689; DSM 40062; HUT 6019; HUT 6053; IFO 12771; IFO 3197; IFO 3412; IFO 3716; IMET 40280; ISP 5062; NRRL B-1329; NRRL B-2685; RIA 1038; RIA 635; VKM Ac-1723
JCM 4133[T]	*Streptomyces fradiae*	ATCC 10745; ATCC 19760; CBS 498.68; CCM 3174; CCRC 12196; DSM 40063; HUT 6095; IFM 1030; IFO 12773; IFO 3439; IFO 3718; IMET 42051; IMI 061202; ISP 5063; JCM 4579; NCIMB 11005; NCIMB 8233; NRRL B-1195; PCM 2330; RIA 1040; RIA 97; VKM Ac-150; VKM Ac-151; V
JCM 4380[T]	*Streptomyces griseobrunneus*	ATCC 19762; CBS 500.68; CCRC 13674; CCUG 11105; DSM 40066; IFO 12775; IMET 42052; ISP 5066; NCIMB 12975; NRRL B-2095; RIA 1042; VKM Ac-753
JCM 4578[T]	*Streptomyces flavovirens*	AS 4.575; ATCC 19758; ATCC 3320; CBS 129.20; CBS 279.30; CBS 496.68; CCM 3243; CCRC 13689; DSM 40062; HUT 6019; HUT 6053; IFO 12771; IFO 3197; IFO 3412; IFO 3716; IMET 40280; ISP 5062; NRRL B-1329; NRRL B-2685; RIA 1038; RIA 635; VKM Ac-1723
JCM 4579[T]	*Streptomyces fradiae*	ATCC 10745; ATCC 19760; CBS 498.68; CCM 3174; CCRC 12196; DSM 40063; HUT 6095; IFM 1030; IFO 12773; IFO 3439; IFO 3718; IMET 42051; IMI 061202; ISP 5063; JCM 4133; NCIMB 11005; NCIMB 8233; NRRL B-1195; PCM 2330; RIA 1040; RIA 97; VKM Ac-150; VKM Ac-151; V
LMG 19371[T]	*Streptomyces fradiae*	ATCC 10745; ATCC 19760; CBS 498.68; CCM 3174; CCRC 12196; DSM 40063; HUT 6095; IFM 1030; IFO 12773; IFO 3439; IFO 3718; IMET 42051; IMI 061202; ISP 5063; JCM 4133; JCM 4579; KCC S-0133; KCC S-0579; Lanoot R-8739; NCIMB 11005; NCIMB 8233; NRRL B-1195; PCM 2330; VKM Ac-152; VKM Ac
NCIMB 8233[T]	*Streptomyces fradiae*	ATCC10745; ATCC19760; CBS498.68; CCM3174; CMI61202; DSM40063; DSM46372; ETH13472; HUT6095; IAM0083; IFO3439; IFO3718; IFO12773; IMET42051; IMRU3535; ISP5063; JCM4133; JCM4579; KCCS- 0133; KCCS-0579; NCIMB11005; NRRL B-1195; RIA97; RIA 1040
NCIMB 12975[T]	*Streptomyces griseobrunneus*	ATCC19762; CBS500.68; DSM40066; HMGBB930; IFO12775; IMET42052; IMRU3068; ISP5066; JCM4380; KCCS-0380; RIA1042
R-8739[T]	*Streptomyces fradiae*	LMG 19371; JCM4579
VKM Ac-1723[T]	*Streptomyces flavovirens*	ISP 5062; RIA 1038; ATCC 3320; ATCC 19758; CBS 129.20; CBS 496.68; CBS 189.75; DSM 40062; IFO 3412; IFO 3716; IFO 12771; JCM 4035; JCM 4578
VKM Ac-150[T]	*Streptomyces fradiae*	ISP 5063; VKM Ac-151; VKM Ac 152; VKM Ac 764; RIA 1040; ATCC 10745; ATCC 19760; CBS 498.68; DSM 40063; IFO 3718; IFO 12773; NCIMB 8233; JCM 4133; JCM 4579

Table 2.4: Example strain class that demonstrates the presence of anomalies in the synonymy evidence collected from different heterogeneous data sources.

based on the minimization of stochastic complexity [32]. Whatever classification method chosen for grouping the binary vectors, after transversal clustering it should be relatively straightforward to delineate subgroups of corresponding rows and columns in the matrix. These subgroups then correspond with the different strains that were merged together into the same strain class, whereas the outliers of the subgroups indicate the exact location and nature of the errors made in the data source records. Note however that irrespective of the level of automation built into the above error detection/correction strategy, our experience learns that manual inspection of the end result is primordial for a decent correction of the errors. An error detection/correction strategy that is solely based on the information in the synonym cross-references tables might not be sufficiently armed to completely unravel the ins and outs of some of the more difficult inconsistencies detected in strain classes of the integrated strain database, e.g. in cases where the same original error has been propagated over several data sources due to the manual copying of the synonym information between data sources. For these complicated cases, the more time-consuming process of integrating the fragmentary strain history [16] has proven to be a suitable alternative for correctly resolving the errors made in the data sources. How these completely integrated strain histories are assembled is further discussed in more detail in subsection 2.3.2.

Figure 2.2 shows a graphical representation of the end-result after performing the previously discussed error detection/correction strategy for resolution of the inconsistencies in the example strain class of Table 2.4. Horizontal and vertical classification was performed using the unweighted pair-group method using arithmetic averages (UPGMA) hierarchical clustering method [67] working upon intermediate pairwise Dice similarity matrices [20] for both the binary row and column vectors of the synonym cross-reference matrix constructed for the strain class. Manual delineation of the different subclasses was then a formality for this example. The transversal clustering indeed points out that three different strains were illegitimately joined together into this strain class, which is completely in agreement with our previous prediction based on the species names assigned to the different data source records. The detected outliers which reflect putative incorrect synonym references, are indicated using red colored boxes in the graphical representation of the synonym cross-references matrix. From these outlying boxes one can logically deduce that the catalogue entry of the culture with label `DSM 40066`$^{\mathsf{T}}$ makes reference to `CBS 498.68` as a synonym label, whereas other data sources more convincingly suggest that the correct synonym should be `CBS 500.68`. Similarly, the catalogue entry `NCIMB 8233`$^{\mathsf{T}}$ should probably refer to `RIA 1040` as a synonym label, instead of making reference to the label `RIA 1038`. With the necessary tools built into the integrated strain database, we could finally split the inconsistent strain class in its semantically correct subclasses which correspond with the type strains of *Streptomyces flavovirens*, *S. fradiae* and *S. griseobrunneus*. Remark also that due to this split action, the evidence completeness w_s^u of the strain classes substantially increases to respectively $\frac{122}{14\times33} \approx 0.2641$ for the *S. flavovirens* type strain, $\frac{251}{21\times46} \approx 0.2598$ for the *S. fradiae* type strain and $\frac{77}{10\times17} \approx 0.4529$ for the *S. griseobrunneus* type strain, and that no classes with mixed species naming are left over after the split procedure.

To conclude with, not only the completeness of the integrated strain database will benefit from the incorporation of as many data sources as possible, but also the data cross-checking

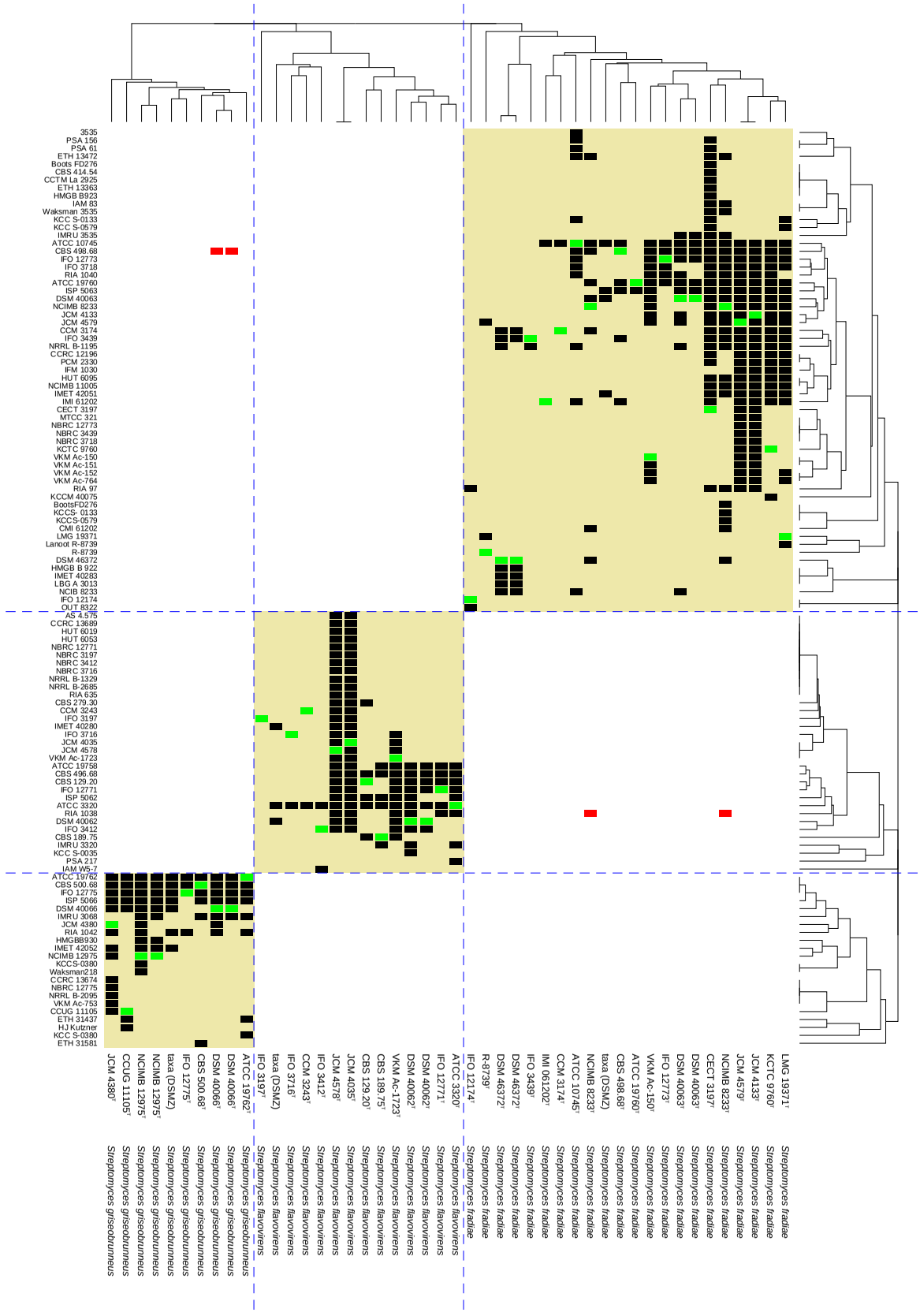


Figure 2.2: Synonym cross-reference matrix that illustrates the procedure followed by the error detection/correction strategy for a strain class where the synonyms of three different type strains have been falsely merged into a single equivalence class, due to errors in the catalogue entries of DSM 40066^{T} and NCIMB 8233^{T}.

power of the previously discussed error detection/correction strategy will significantly improve if more pieces of overlapping evidence become available. The current restricted scope of the integrated strain database caused by the limitation of merely processing only bacterial data sources, precludes that erroneous cross-references between labels of strains of bacteria, fungi and yeasts can be detected or corrected. However, these errors have already infected the integrated strain database because some data sources provide mixed synonym information on labels of all these different kinds of microorganisms. Monge and Elkan [56] claim that for most duplicate detection problems a small number of false positives and false negatives can be tolerated. They also estimate that in most heterogeneous database systems there are only a few number of errors, so that the equivalence relations found by a performant deduplication algorithm will be a good approximation of the true semantic equivalence classes. By application of the error detection/correction strategies outlined in this section for scanning the correctness of the equivalence classes of the integrated strain database, the results of the data quality assessment made in the next section will clearly demonstrate that the amount of errors made in the microbial data sources cannot be neglected during the construction of solid information systems.

2.3.2 Integrated strain history

The information captured within the synonym cross-reference matrices alone may not be sufficient to completely unravel the ins and outs of the more complex inconsistencies within the integrated strain database. As an alternative approach for error detection and correction of synonym equivalences, integration of the fragmented strain history information into a complete history tree has proven to be a very clarifying tool for resolving most of the more complicated cases. In this subsection we will discuss some of the efforts required for the automatic construction of complete strain history trees and illustrate their error detection capabilities through one of the striking examples encountered during curation of the integrated strain database.

Culture collection catalogues record the strain history information within a field called `History of Deposit` according to the CABRI standard [9], which should describe the history of the cultured sample from its deposit into the collection up to the initial point of isolation. All deposit history information for a given strain can then be easily extracted from the online data sources linked to the corresponding strain class in the integrated strain database. As an example, Table 2.5 shows the history information for the *Bacillus cereus* type strain as it can be found in the catalogues of a number of culture collections. This table clearly demonstrates that the different data sources have adopted different formats for encoding the strain history information, due to the fact that the CABRI standard does not give a formal prescription of formatting the content of the `History of Deposit` field. As a consequence, it is a daunting task for software agents to process the history information in a fully automated way. Nevertheless, the history information of the *Bacillus cereus* type strain in Table 2.5 contains quite some duplication, what makes that after standardization and normalization of the data, the complete strain history tree of the *Bacillus cereus* type strain can be represented in a more informative way, as is shown in Figure 2.3. These kind

Source	Catalog entry	Species name	History
ATCC	ATCC 14579^{T}	*Bacillus cereus*	ATCC<<–RE Gordon <<–T. Gibson 971 <<— W. Ford 13
CABRI	CIP 66.24^{T}	*Bacillus cereus*	ATCC 1966 < R.E. Gordon: strain NRRL B-3711 < T. Gibson:strain 971 < W.W. Ford: strain 13
CABRI	DSM 31^{T}	*Bacillus cereus*	<- ATCC <- R.E. Gordon <- T. Gibson, 971 <- W.W. Ford, 13
CABRI	LMD 75.8^{T}	*Bacillus cereus*	LMD < Jun 1975, ATCC < R.E. Gordon < T. Gibson < W. Ford
CABRI	LMG 6923^{T}	*Bacillus cereus*	<- 1985, DSM <- ATCC <- R.Gordon <- T.Gibson <- W.Ford
CABRI	NCIMB 9373^{T}	*Bacillus cereus*	T.Gibson – W.W.Ford
CCM	CCM 2010^{T}	*Bacillus cereus*	R.E. Gordon <- T. Gibson <- W.W. Ford
CCRC	CCRC 10603^{T}	*Bacillus cereus*	10603<< ATCC << R. E. Gordon << T. Gibson 971 << W. Ford 13
CCRC	CCRC 11026^{T}	*Bacillus cereus*	11026<< IAM
CCUG	CCUG 7414^{T}	*Bacillus cereus*	M.Kocur,CCM,Brno,Czechoslovakia 10 Aug 1978 <R.E.Gordon,IMRU<T.Gibson<W.W.Ford
CECT	CECT 148^{T}	*Bacillus cereus*	CECT, 1974 < NCTC, 1963 < T. Gibson < W.W. Ford.
CECT	CECT 5050^{T}	*Bacillus cereus*	CECT, 1992 < DSMZ < ATCC < R.E. Gordon < T. Gibson < W.W. Ford.
CIP	CIP 66.24^{T}	*Bacillus cereus*	1966, ATCC <– R.E. Gordon: strain NRRL B-3711 <– T. Gibson: strain 971 <– W.W. Ford: strain 13
DSM	DSM 31^{T}	*Bacillus cereus*	<- ATCC <- R.E. Gordon <- T. Gibson, 971 <- W.W. Ford, 13
IFO	IFO 15305^{T}	*Bacillus cereus*	1992. JCM 2152 <== IAM 12605 <== NCIB 9373 <== R.E. Gordon
JCM	JCM 2152^{T}	*Bacillus cereus*	<– IAM 12605 <– NCIB 9373 <– R. E. Gordon
KCTC	KCTC 3624^{T}	*Bacillus cereus*	<- IFO <- JCM <- IAM <- NCIB <- R.E. Gordon
LMG	LMG 6923^{T}	*Bacillus cereus*	<- 1985, DSM <- ATCC <- R.Gordon <- T.Gibson <- W.Ford
UKNCC	NCIMB 9373^{T}	*Bacillus cereus*	T.Gibson – W.W.Ford

Table 2.5: Strain history information of the *Bacillus cereus* type strain, as it was found in different catalogues of culture collections that are available online.

of graphs can be flexibly drawn with assistance of the `dot` software package [46]. In this graphical representation of the completely integrated strain history tree, the orange boxes represent cultures of the *Bacillus cereus* type strain as they are stored in different culture collections or private research collections. These boxes are labeled with the strain number assigned by the collection holder. The role of culture equivalence classes for the representation of syntactical variation in the spelling of labels is clearly visible in Figure 2.3, where the labels `13` and `Ford 13` form a single culture node in the history tree. *Idem dito* for the labels `971`, `Gibson` and `Gibson 971`. If the collection holder additionally provides some identification of the cultured sample, the identification is shown in the bottom half of the orange box. Deposit of a culture from one collection into another is represented by an arrow linking the corresponding boxes, and annotated with the information known about this deposition (date of deposit, depositor name, depositor institute,...). Labels from the *Bacillus cereus* type strain equivalence class that have an unknown position within the completely integrated strain history tree, were lumped into a single blue box in the upper left corner of Figure 2.3. More examples can be found in Appendix A, which contains the completely integrated strain history of all *Enterococcus* spp. type strains.

According to the synonym label equivalences found for the *Enterococcus gallinarum* type strain in different online culture collection catalogs, all strain numbers mentioned in the graphical representation of the complete strain history in Figure 2.4 constitute a single equivalence class. However, several catalog entries (`NCTC 11428`, `ATCC 35038`, `LMG 11207`) contain some direct or indirect evidence that the corresponding cultures should possibly be identified as *Enterococcus faecalis*. Within the history tree, one could interprete this as a putative contamination of the complete branch rooted at the node labeled NCTC 11428 (or any higher node). The nodes of this affected branch are coloured

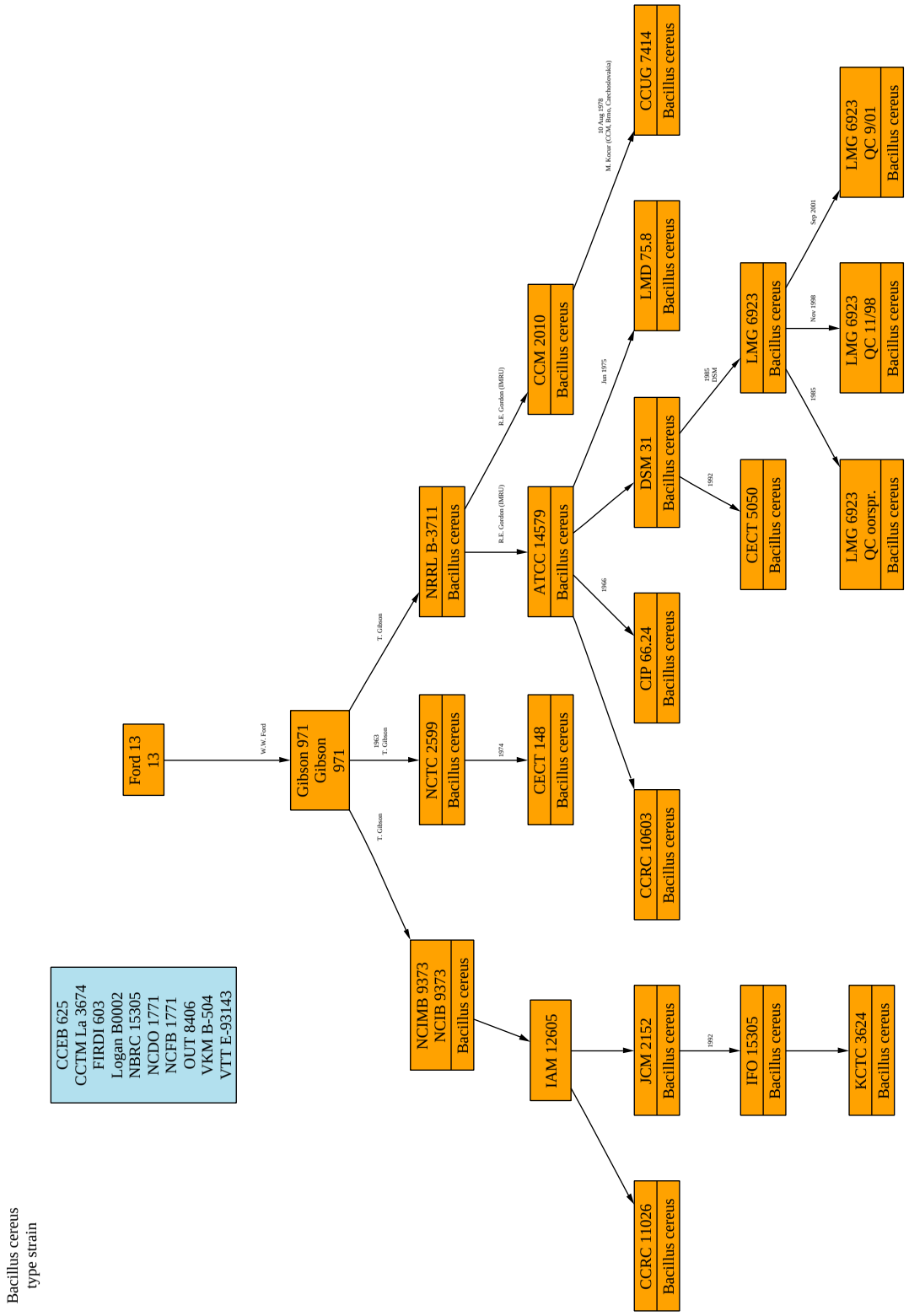


Figure 2.3: Completely integrated strain history tree of the *Bacillus cereus* type strain.

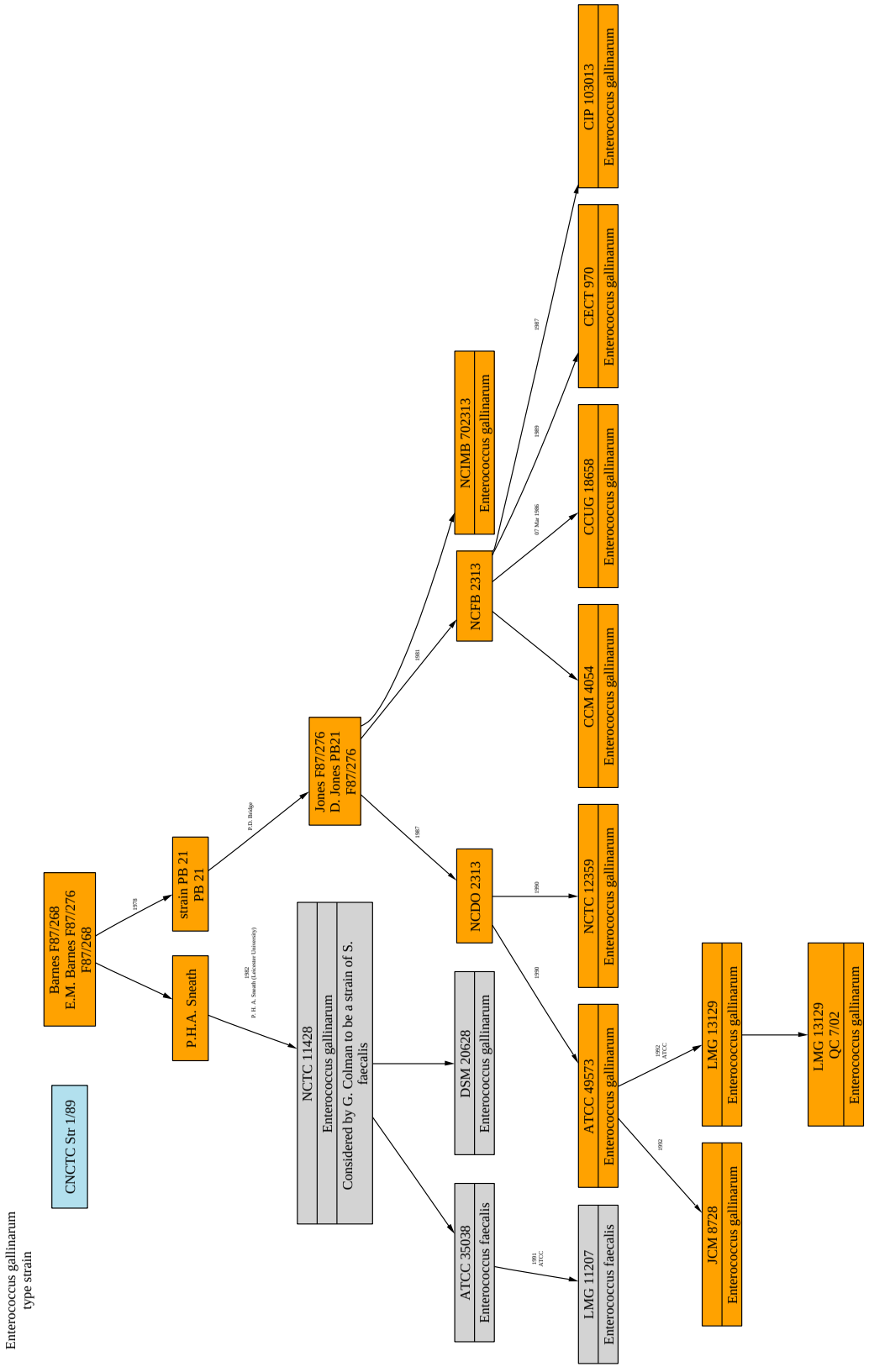


Figure 2.4: Complete strain history tree of the *Enterococcus gallinarum* type strain, showing a putative contamination within the light gray coloured branch. Several sources indicate that the cultures of the affected branch should be identified as *Enterococcus faecalis*.

light gray in the complete tree representation of Figure 2.4. A polyphasic identification based on the analysis of fatty acid composition, sodium dodecyl sulphate (SDS) polyacrylamide gel electrophoresis (PAGE), fluorescent amplified fragment length polymorphism (fAFLP) gel electrophoresis, and the complete DNA sequences of the atpA, pheS and rpoA genes provides enough evidence that at least the culture `LMG 11207` harboured in the BCCM™/LMG Bacteria Collection indeed belongs to the species *Enterococcus faecalis*. Comparable emperical information for the other cultures of the defective branch is required in order to definitively sort out the nature and a possible solution for the anomalies in the information provided by the different culture collections for this case.

Apart from playing a significant role within the correction of errors in the equivalence classes of synonym labels, integrated strain history knowledge also is an important prerequisite for tracking and tracing the tranfer of microbial genetic resources (MGRs), in light of monitoring intellectual property right issues when identifying the individuals and groups that are entitled to be scientifically or financially rewarded for their contribution to the conservation and sustainable use of the MGRs. This follows the conditions stated in the Convention on Biological Diversity (Rio de Janeiro, June 5th, 1992).

2.4 Data quality assessment

Microbiologists isolate new strains on a daily basis, and cultures of existing strains are regularly transferred between instances for a number of different purposes. As such, the amount of equivalence information about the labels used in the field of microbiology grows continuously, which turns the problem of learning and maintaining knowledge on the existing equivalences into an ever ongoing process. In this section we will therefore describe some of the properties of label equivalences, measured on a single point-in-time snapshot taken during the lifetime of the information accumulated in the integrated strain database $\mathcal{I}$. Nonetheless, this will allow us to draw some general trends and conclusions.

In order to compose a complete picture of all the existing equivalences concerning the labels assigned to microorganisms, it is evident that we need to enclose as many sources of information as possible for the construction of a central repository. Table 2.6 lists the data sources processed by the `IncrementEquivalence` procedure, at the stage of the integration process on which the snapshot of $\mathcal{I}$ was taken. The wide range of document exchange formats currently adopted by the different data sources, forms a major obstacle for automation of the data collection process. Consequently, specific mediating agents for retrieving all documents provided by the data sources and specific parsers for extracting synonym label records and other relevant information from these documents, needed to be separately implemented within the integrated strain database for practically every data provider. Development of and adherence to worldwide accepted document formatting standards for exchange of microbial information, such as MINE [30, 70], CABRI [9], Darwin Core [15] or ABCD [1], would dramatically simplify this processing step. Some data sources that originate from the same instance are available online in multiple versions. For example, the `LMG` catalogue is both available from the BCCM™ website and from the

data source	acronym	URL
American Type Culture Collection	ATCC	www.atcc.org
Czech Collection of Microorganisms	CCM	www.sci.muni.cz/ccm
Culture Collection, University of Göteborg, Sweden	CCUG	www.ccug.gu.se
Colección Española de Cultivos Tipo	CECT	www.cect.org/english/index.htm
Collection de l'Institut Pasteur	CIP	www.pasteur.fr/externe
Deutsche Sammlung von Mikroorganismen und Zellkulturen	DSMZ	www.dsmz.de
Institute for Fermentation, Osaka	IFO	www.ifo.or.jp/index_e.html
Japan Collection of Microorganisms	JCM	www.jcm.riken.go.jp
Korean Collection for Type Cultures	KCTC	kctc.kribb.re.kr/english
Laboratory of Microbiology, Ghent	LMG	www.belspo.be/bccm/lmg.htm
Pasteur Culture Collection of Cyanobacteria	PCC	www.pasteur.fr/recherche/banques/PCC
All-Russian Collection of Microorganisms	VKM	www.vkm.ru
Common Access to Biological Resources and Information	CABRI	www.cabri.org
CABRI collects several culture collection catalogs into a uniform online catalogue. Acronyms of bacterial subcatalogs currently covered within integrated strain database: CBS, CIP, DSMZ, IMI, LMD, LMG, MUCL, NCIMB		
United Kingdom National Culture Collection	UKNCC	www.ukncc.co.uk/index.htm
UKNCC collects several culture collection catalogs into a uniform online catalogue. Acronyms of subcatalogs currently covered within the integrated strain database: NCIMB, NCPPB, NCTC		
Bacterial Nomenclature Up-to-Date (DSMZ)	taxa (DSMZ)	www.dsmz.de/bactnom/bactname.htm

Table 2.6: Data sources currently contributing to the equivalence relations covered within the integrated strain database.

CABRI portal, while the NCIMB catalogue is incorporated in both the CABRI and UKNCC suites, as shown in Table 2.6. Because the information content of different online versions of the same catalogue is not necessarily identical, we have treated each version as a separate data source in the discussion here. Additionally, the information content of a given data source is sensitive for changes over time, as new strains or supplementary synonyms are included. We may however suppose that the information content of each earlier snapshot of a data source is a subset of the information content of any more recent snapshot of the data source. Hence, we have treated the last processed snapshot of a data source as the one that provides all the information content of that data source and obsoletes all previous snapshots. The instance acronyms specified in Table 2.6 are used as a reference for the data sources in the rest of this chapter, where data sources that are embedded in the CABRI suite are marked with a $^{\mathsf{c}}$ superscript and data sources from the UKNCC suite with a $^{\mathsf{u}}$ superscript. Note that the data sources currently processed for the construction of the integrated strain database $\mathcal{I}$ primarily contain information on bacterial strains. Data sources that cover strains of fungi and yeasts are planned to be incorporated for the generation of future versions of the database.

The current version of the integrated strain database is constructed within an Oracle 8.1.7 database (Oracle corporation, CA, USA) using the Data Warehousing technology [49], and contains information on $|\mathcal{C}| = 311410$ cultures which cluster together in a total of $|\mathcal{S}| = 120695$ strain classes. For every set of labels $L \subseteq \mathcal{L}$, we denote the subset that contains all unique labels of L as $U(L)$. With $\mathcal{S}^{\mathsf{T}}$ we denote the subset of $\mathcal{S}$ that corresponds to the official list of type strains defined in bacterial taxonomy, based on their valid publication in the International Journal of Systematic and Evolutionary Microbiology. An electronic version of this list is maintained by the Deutsche Sammlung von Mikroorganismen und Zellkulturen, and published online under the headings of Bacterial Nomenclature Up-to-date [73]. This data source has also been included in Table 2.6 as the data source with acronym taxa (DSMZ). The average number of unique labels assigned to all strains in

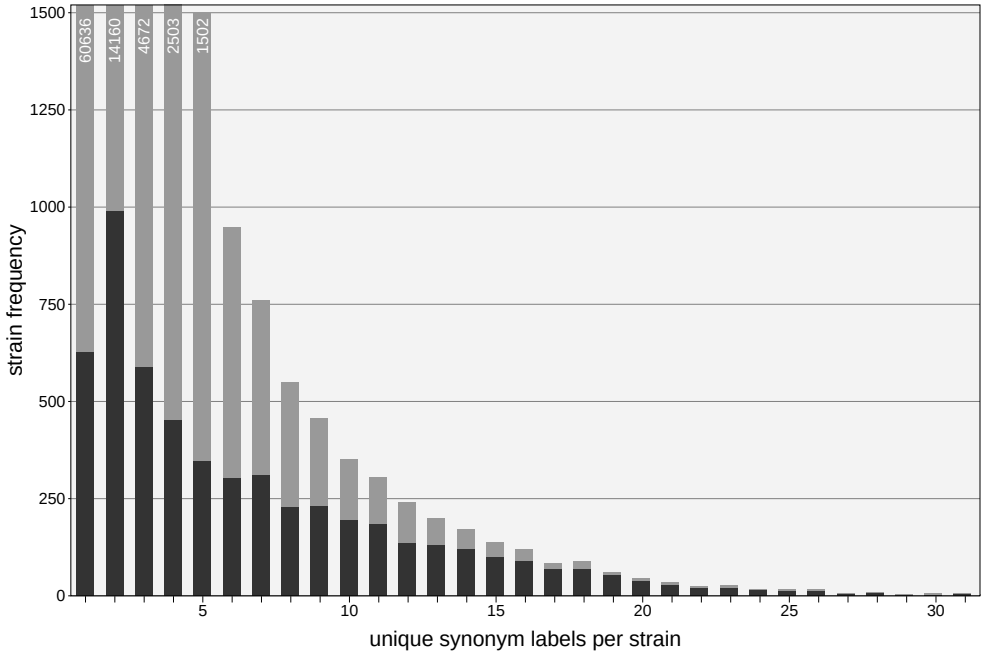


Figure 2.5: Histogram of the amount of unique synonym labels per strain $|U(s)|$ for all strains $s \in \mathcal{S}$.

$\mathcal{S}$ is 1.919 ($\pm$ 2.477), while the average number of unique labels assigned to the type strains in $\mathcal{S}^{\mathsf{T}}$ is 6.762 ($\pm$ 6.092). This clearly indicates that in general type strains are more wide-spread than other strains, which is logical given their crucial role in the taxonomy of bacteria. Figure 2.5 shows the histogram of the number of unique synonym labels per strain $|U(s)|$ for all strains $s \in \mathcal{S}$. The contribution of the type strains has been colored dark gray in this histogram, while a lighter shade of gray was used for the contribution of the other strains. The number of unique labels $|U(s)|$ known for a strain $s \in \mathcal{S}$ can be considered as a measure for the global spread of the strain, hence also for the popularity of the strain for use in biological applications. With this in mind, Table 2.7 depicts a list of the most popular strains in the integrated strain database, retrieved as the strains $s \in \mathcal{S}$ with $|U(s)| \geq 35$. In this table we have chosen one of the synonymous labels of each strain as the representative label for referencing the strain. Note also that it is common practice in bacteriology that labels of type strains are annotated with a superscript $^{\mathsf{T}}$. Table 2.8 presents the amount of overlap between the microbial organisms deposited in a selection of culture collections. The diagonal elements of the depicted matrix represent the number of strains in the integrated strain database that are stored in the corresponding culture collection, while the non-diagonal elements given the number of strains that are harboured in both collections indicated at the left and bottom of the rows and columns in the matrix. From the list of popular strains in Table 2.7 and the histogram shown in Figure 2.5 it should be clear that the set of type strains $\mathcal{S}^{\mathsf{T}}$ can be regarded as the greatest common divisor of all bacterial data sources. For the description of some further statistics of the integrated strain database and its composing data sources, we have therefore restricted the scope to the set of type strains $\mathcal{S}^{\mathsf{T}}$, with a further limitation to the unique labels of these type strains wherever this was appropriate.

label of s	taxon	$\lvert U(s)\rvert$
LMG 6326$^{\mathsf{T}}$	*Bacillus coagulans*	35
LMG 16798$^{\mathsf{T}}$	*Bacillus lentus*	35
LMG 13550$^{\mathsf{T}}$	*Lactobacillus acidophilus*	36
LMG 8195	*Staphylococcus aureus* subsp. *aureus*	36
LMG 7135$^{\mathsf{T}}$	*Bacillus subtilis* subsp. *subtilis*	36
LMG 5973$^{\mathsf{T}}$	*Streptomyces griseocarneus*	37
LMG 6400$^{\mathsf{T}}$	*Lactobacillus rhamnosus*	38
LMG 8221	*Bacillus cereus*	39
LMG 6399$^{\mathsf{T}}$	*Enterococcus hirae*	40
LMG 7558	*Bacillus licheniformis*	41
LMG 4049$^{\mathsf{T}}$	*Paracoccus denitrificans*	41
LMG 4050$^{\mathsf{T}}$	*Micrococcus luteus*	42
LMG 1242$^{\mathsf{T}}$	*Pseudomonas aeruginosa*	42
LMG 5359$^{\mathsf{T}}$	*Rhodococcus erythropolis*	43
LMG 19302$^{\mathsf{T}}$	*Streptomyces griseus* subsp. *griseus*	44
LMG 19371$^{\mathsf{T}}$	*Streptomyces fradiae*	44
LMG 5968$^{\mathsf{T}}$	*Streptomyces aureofaciens*	44
LMG 13261$^{\mathsf{T}}$	*Bacillus circulans*	45
LMG 8197	*Bacillus subtilis* subsp. *spizizenii*	48
LMG 2189$^{\mathsf{T}}$	*Pseudomonas fluorescens*	49
LMG 1284	*Lactobacillus plantarum*	51
ATCC 8664$^{\mathsf{T}}$	*Streptomyces lavendulae* subsp. *lavendula*	52
LMG 8596$^{\mathsf{T}}$	*Streptomyces rimosus* subsp. *rimosus*	52
LMG 16000$^{\mathsf{T}}$	*Brevibacillus laterosporus*	53
LMG 1673$^{\mathsf{T}}$	*Gluconobacter oxydans* subsp. *suboxydans*	56

Table 2.7: List of popular strains, determined as the strains $s \in \mathcal{S}$ with $|U(s)| \geq 35$.

	ATCC	CCM	CCUG	CECT	CIP	DSM	IFO	JCM	KCTC	LMD	LMG	NCIMB	NCPPB	NCTC
ATCC	19090													
CCM	1072	2526												
CCUG	2418	815	15148											
CECT	1567	366	729	3682										
CIP	2667	812	2102	729	7432									
DSM	4706	866	1677	1009	2140	11873								
IFO	3627	432	516	750	778	2111	11924							
JCM	3180	511	932	690	1275	2654	2701	7030						
KCTC	2652	350	705	658	853	1712	1543	1615	5191					
LMD	316	130	148	119	154	188	118	86	104	1006				
LMG	2986	858	2896	870	1875	2231	987	1170	899	269	17369			
NCIMB	2665	664	986	794	1299	2100	1108	1259	966	251	1722	7123		
NCPPB	378	69	120	84	115	138	44	50	34	10	1414	35	3184	
NCTC	1706	450	1266	518	1143	738	324	467	350	120	856	661	31	5030

Table 2.8: Common strain statistics for a selection of culture collections.

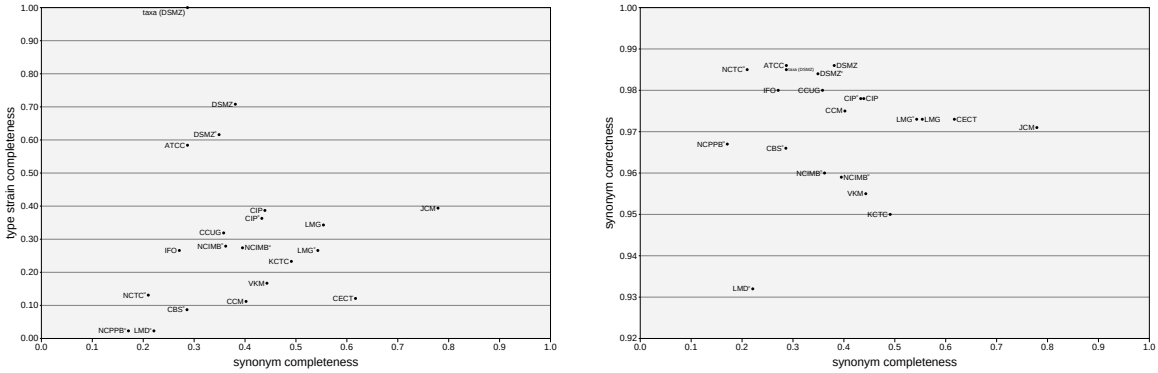


Figure 2.6: Scatterplots of strain completeness versus synonym completeness versus synonym correctness, for all type strains included in the integrated strain database.

For a given data source d, the subset of records that correspond with a type strain in $\mathcal{S}^{\mathsf{T}}$ is indicated as d^{T}, thus formally we have that

$$d^{\mathsf{T}} \equiv \{r \in d \,|\, \mathcal{S}_r \in \mathcal{S}^{\mathsf{T}}\} \,. \tag{2.14}$$

The set of strains covered by the data source d is noted as $\mathcal{S}_d$, with

$$\mathcal{S}_d \equiv \{\mathcal{S}_r \in \mathcal{S} \,|\, r \in d\} \,. \tag{2.15}$$

We use the short notation $\mathcal{S}_d^{\mathsf{T}} \equiv \mathcal{S}_d \cap \mathcal{S}^{\mathsf{T}}$, to indicate the set of type strains covered by the data source d. For most data sources d that were processed for the construction of the integrated strain database $\mathcal{I}$, we have shown the number of records $|d|$, the number of type strain records $|d^{\mathsf{T}}|$, and the number of type strains covered $|\mathcal{S}_d^{\mathsf{T}}|$ in Table 2.9. From this table it is immediately obvious that different online versions that originate from the same instance database do not necessarily cover the same information content. A comparison of the number of type strain records and the number of type strains covered by a given data source indicates that there is some degree of overlap within data sources, in that some records refer to the same type strain. Our experience learns that the synonym information is not necessarily the same for overlapping records that belong to the same data source. This can easily be seen within the cross-reference table shown in Figure 2.1, where the synonym information provided by the records `CECT 148`$^{\mathsf{T}}$ and `CECT 5050`$^{\mathsf{T}}$ is not identical, although both records are extracted from the same data source and represent different cultures of the same strain. Moreover, the record that corresponds with label `CECT 148`$^{\mathsf{T}}$ fails to refer to its synonym record `CECT 5050`$^{\mathsf{T}}$, whereas the reverse synonym equivalence reference is indeed correctly provided by the `CECT` data source.

With the knowledge accumulated in the integrated strain database $\mathcal{I}$, we can make an evaluation of the completeness and correctness of the information content of the different data sources. As an estimation of the completeness of the strain information provided by a given data source d, we define the *type strain completeness* γ_d as the fraction of type strains covered by that data source, with respect to the number of type strains covered within the integrated strain database

$$\gamma_d = \frac{|S_d^{\mathsf{T}}|}{|\mathcal{S}^{\mathsf{T}}|} \,. \tag{2.16}$$

d	$\|d\|$	$\|d^{\mathsf{T}}\|$	$\|\mathcal{S}_d^{\mathsf{T}}\|$	γ_d	σ_d	ϵ_d
ATCC	16408	3740	3157	0.584	0.287	0.986
CBS$^{\mathsf{c}}$	1038	503	472	0.087	0.286	0.966
CCM	2214	608	604	0.112	0.402	0.975
CCUG	15401	2045	1722	0.319	0.358	0.980
CECT	3578	665	653	0.121	0.617	0.973
CIP	7181	2122	2090	0.387	0.439	0.978
CIP$^{\mathsf{c}}$	7038	1989	1962	0.363	0.433	0.978
DSMZ	11378	3885	3826	0.708	0.381	0.986
DSMZ$^{\mathsf{c}}$	8067	3378	3327	0.616	0.349	0.984
IFO	11834	1496	1439	0.266	0.271	0.980
JCM	7123	2369	2132	0.394	0.779	0.971
KCTC	5465	1445	1262	0.233	0.491	0.950
LMD$^{\mathsf{c}}$	1016	147	122	0.023	0.221	0.932
LMG	14780	2136	1852	0.343	0.554	0.973
LMG$^{\mathsf{c}}$	11912	1510	1439	0.266	0.543	0.973
NCIMB$^{\mathsf{c}}$	6921	1555	1508	0.279	0.362	0.960
NCIMB$^{\mathsf{u}}$	14051	2975	1481	0.274	0.395	0.959
NCPPB$^{\mathsf{u}}$	3113	126	126	0.023	0.171	0.967
NCTC$^{\mathsf{u}}$	4934	732	706	0.131	0.210	0.985
VKM	2196	917	903	0.167	0.443	0.955
taxa (DSMZ)	8014	6475	5405	1.000	0.287	0.985

Table 2.9: Completeness and correctness statistics for some data sources that were incorporated during the construction of the integrated strain database $\mathcal{I}$. Data sources marked with $^{\mathsf{c}}$ are subsections of the CABRI suite, while data sources marked with $^{\mathsf{u}}$ are part of the UKNCC suite.

Alternatively, we define the *synonym completeness* σ_d of the data source d as a comparison between the number of unique synonym labels for type strains provided by the data source, and all unique synonym labels for type strains known in the integrated strain database $\mathcal{I}$, in the following way

$$\sigma_d = \frac{\sum_{r \in d^{\intercal}} |U(r)|}{\sum_{r \in d^{\intercal}} |U(\bar{r})|} \,. \tag{2.17}$$

Herein, the *completion* $\bar{r}$ of a record r represents the set of all synonym labels known within the integrated strain database $\mathcal{I}$ for the strain that corresponds with that record, thus

$$\bar{r} = \{ l \in \mathcal{L} \,|\, \mathcal{S}_l = \mathcal{S}_r \} \,. \tag{2.18}$$

The type strain completeness and synonym completeness values calculated for the data sources mentioned in Table 2.9, clearly demonstrate that in general the equivalence information provided by the different data sources is only partial in comparison with the information content covered within the integrated strain database, both in respect to the number of incorporated strains and the number of synonym labels known for each strain. Only two culture collections house over half of the validly described type strains, being `DSMZ` with 71% of the type strains and `ATCC` with 58% of the type strains. On the other hand, Table 2.9 also indicates that only three culture collections manage to disseminate over half of the unique labels known by the integrated strain database, being `JCM` with 78%, `CECT` with 62% and `LMG` with 55%. Data sources that have a high score for type strain completeness, generally score worse for synonym completeness, and vice versa, as can be seen from the left scatterplot in Figure 2.6. To conclude with, no single culture collection harbours cultures of all type strains, and all data sources score only moderately or low for the completeness of the synonym labels they mention of the strains they cover. These observations confirm the need for a central repository that maintains updated information for all synonym labels known of all strains used in microbiology.

If we denote by $d^{\dagger}$ the subset of records of a given data source d wherein at least one inconsistency was detected by the error detection/correction strategy discussed in section 2.3, then the *synonym correctness* ϵ_d of a data source d can be measured as

$$\epsilon_d = 1 - \frac{|d^{\dagger\intercal}|}{|d^{\intercal}|} \,. \tag{2.19}$$

Note that in this definition we have used the abbreviated notation $d^{\dagger\intercal}$ for $d^{\dagger} \cap d^{\intercal}$. We have again worked on the limited scope of the type strain records for the calculation of the correctness statistic. We could safely do this without losing the generality of this statistic, because 55% of the strains that were affected by the inconsistencies we have detected were type strains. The synonym correctness values calculated for the different data sources in Table 2.9 indicate that the amount of errors made in each data source is relatively low, as the total number of affected type strain records ranges from 1.4% to 6.8% for the inspected data sources. However, the fact that a single error against the pairwise definition of semantic equivalence may cause a cascade of other equivalence errors, is a known property of the transitive closure algorithm that was used to enforce transitivity on the partially defined equivalence relation in section 2.2.2. This phenomenon is known as the *'garbage in, more garbage out'*-effect caused by the transitive closure. As a result, a total of 614

type strain equivalence classes — 11.4% of the currently described type strains — would have been affected by the inconsistencies present in the originating data sources, without the implementation of an error detection/correction strategy such as the one described in section 2.3. This huge amount of errors would be unacceptable for a central repository that envisions both completeness and correctness of the information it disseminates, and it stresses the importance of all the efforts put into the detection and correction of errors during the construction process of the integrated strain database.

Because each of the data sources still autonomously functions as an independent information provider, and synonym information is frequently copied manually from one data source into another, the detected errors should preferably be corrected in these primary sources of information. With this in mind, the errors that were detected and corrected in the integrated strain database $\mathcal{I}$ can be regularly sent as feedback information to the database managers of the orginating data sources. After all, the very nature of synonymy in strain labels implies that a strategy for the detection and correction of errors within a given data source can only discover inconsistencies when looking across the borders of the data source at hand. In other words, this kind of problem cannot be resolved locally. Apart from being used for reporting errors and proposing remedies to the administrators of the inspected data sources, the list of detected and/or corrected errors is also applied in the integrated strain database as a sort of blacklist, in order to prevent that records which contain pairs of labels that occur on the blacklist give rise to the illegitimate merge of strain classes that are known to be distinct. Instead, an exception is launched so that an appropriate evaluation of the anomaly can be made by the administrator of the integrated strain database. As such, this blacklist avoids that errors which were resolved in the integrated strain database, perturbate the equivalence relation again during subsequent handlings of updated data sources wherein previously detected errors were not yet corrected. An additional advantage of the blacklist is that it automatically arms the integrated strain database against the propagation of errors in the data sources, due to the manual copying of synonym labels from one data source into another. Consequently, the list of detected and corrected errors can be regarded as an addendum of negative pairwise equivalences to the equational theory that was discussed in section 2.2.

2.5 Linking autonomous microbial data sources

With the advent and the rapid emergence of the Internet, world wide access to multiple public microbial information services has given a strong impetus to research in the field of microbiology, by instantly disseminating the latest breakthroughs and insights within the problem domain and establishing in the long term a pool of the microbiologist's collective knowledge. The online catalogues of biological resource centers (BRCs) provide basic information on the isolation, identification and availability of many important and well-characterized microorganisms. Emperical databases contain information on many of the genotypic and phenotypic traits of these microbial strains for a broad range of experimental techniques, which seriously vary in their interlab standardization and reproducibility. The International Nucleotide Sequence Database (DDBJ/EMBL/GenBank) [40] has emerged as

one of the greatest successes in the accumulation of reproducible experimental information, providing parts or the whole genetic map of many of the life forms on earth. Completely new branches of research, such as computational genomics, have been established on the foundations of these sequence databases. Finally, probably the largest contribution to microbial research is currently only published in the scientific literature, which in itself forms a heterogeneous knowledge base that is progressively accessible in electronic form.

The bewildering proliferation of these massive amounts of information urges the establishment of solid cross-references between the different autonomous and heterogeneous data sources, in order to reduce the amount of data duplication between the information providers, assist the researchers in the navigation through this data-rich environment by merging all relevant information into a uniform view, and monitor the overall data quality provided by the different web services through continuous cross-checking of the information. Primordial to the establishment of durable cross-reference scenarios is the availability of unique object identifiers for the unequivocal discrimination of and reference to the different entities in the problem domain. As an example, the assignment of accession numbers as the unique object identifiers for genetic sequence data and the use of PubMed identifiers that act on behalf of the scientific publications, have enabled a cross-referencing scheme that maintains mutual links between the International Nucleotide Sequence Database and the large PubMed literature repository collecting scientific publications from the life sciences. However, as a consequence of the lack of unique identifiers for microorganisms, microbial source information has never been involved in similar cross-references. Instead, the necessary information about the microbial samples is partially copied into the peripheral data sources, which perturbates the management of this information that is also subjected to dynamic changes. In this section we will therefore demonstrate how the integrated strain database may form the cornerstone of a solid and manageable cross-referencing system that establishes mutual links between the information provided by biological resource centers, emperical knowledge bases and scientific research papers.

2.5.1 Managing cross-references between BRCs and EMBL

The International Nucleotide Sequence Database stands model for many other databases containing empirical data on microorganisms. Therefore, we have opted for this knowledge base to exemplify how the genetic sequence information can be automatically linked with the natural resources from which the DNA was extracted, and vice versa. In specific, we make use of the European Molecular Biology Laboratory (EMBL) gateway as an access point to the results of the many international initiatives for collecting genomic information (`http://srs.ebi.ac.uk`). The issue at stake is sketched in Figure 2.7, illustrating the desire to cross-reference the genomic sequence records with their associated biological sample records found in the online catalogues of biological resource centers.

A serious deficiency of the public sequence databases is that there is no consistent recording of the label for the individual culture from which the sequence was obtained. According to the EMBL specifications, this information should be stored in the qualifiers

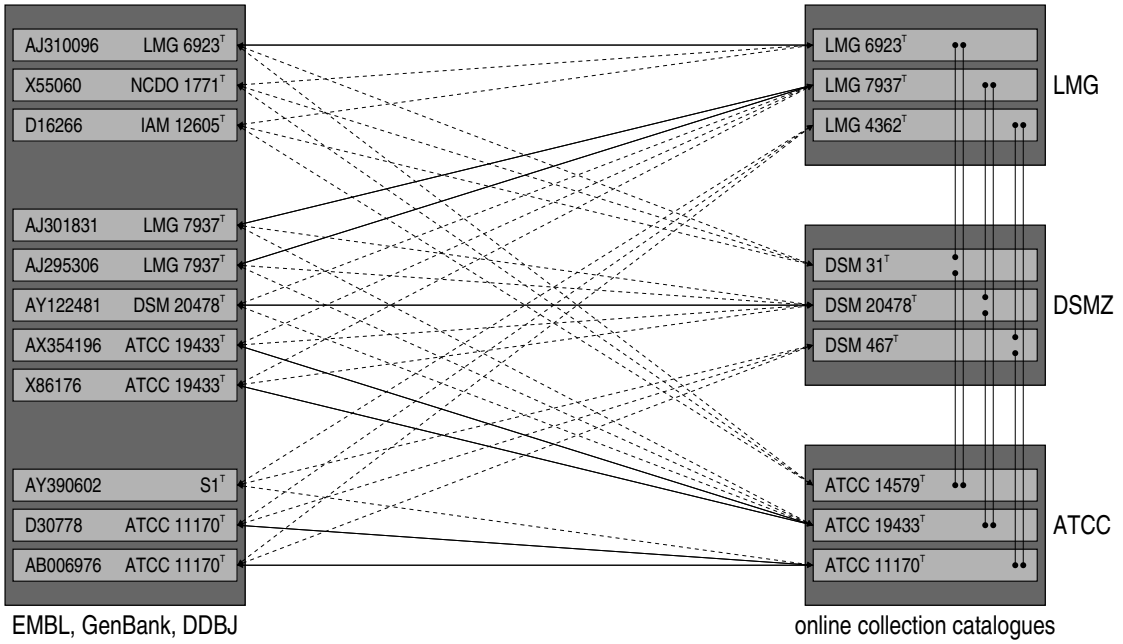


Figure 2.7: Establishing direct or indirect cross-references between biological resource centers and peripheral information sources leads to the cumbersome requirement of maintaining a many-to-many relationship.

`isolate` or `strain` of the `source` feature, but the sequence deposit procedures do not strictly prohibit that depositors provide the strain label information within any of the other fields or – more critical – do not provide this information at all. Therefore, we have developed a software tool that automatically parses complete EMBL formatted sequence records for extraction of the associated strain label information. Missing labels require a time-consuming manual lookup within the literature references associated to the sequence records.

At first sight, the strain labels may seem good candidates for building solid links between the sequence records and the corresponding culture collection catalogue records, but unfortunately these labels associated with biological samples show some form of ambiguity as was discussed in section 2.2. In order to see the consequences of synonymy and homonymy of strain labels on the linkage problem, we refer once again to Figure 2.7. This graphical representation depicts a simplified version of the International Nucleotide Sequence Database on the left side, containing only 11 representative sequence records. Each sequence record is identified by its accession number on the left, whereas the strain label that is extracted from the sequence record is shown on the left-hand side. Note that some strain labels occur in multiple sequence records within the sequence database, and that according to the synonymy of the labels, the example sequences logically belong to three different strains. The right-hand side of the figure shows how sample cultures for each of these strains are harboured in three representative culture collections. The culture collection records are identified by the labels assigned by the instances to each of the samples. In case a cross-referencing scheme is envisioned where the records with corresponding labels are inter-linked, taking into account syntactic variations in the spelling of the labels, one

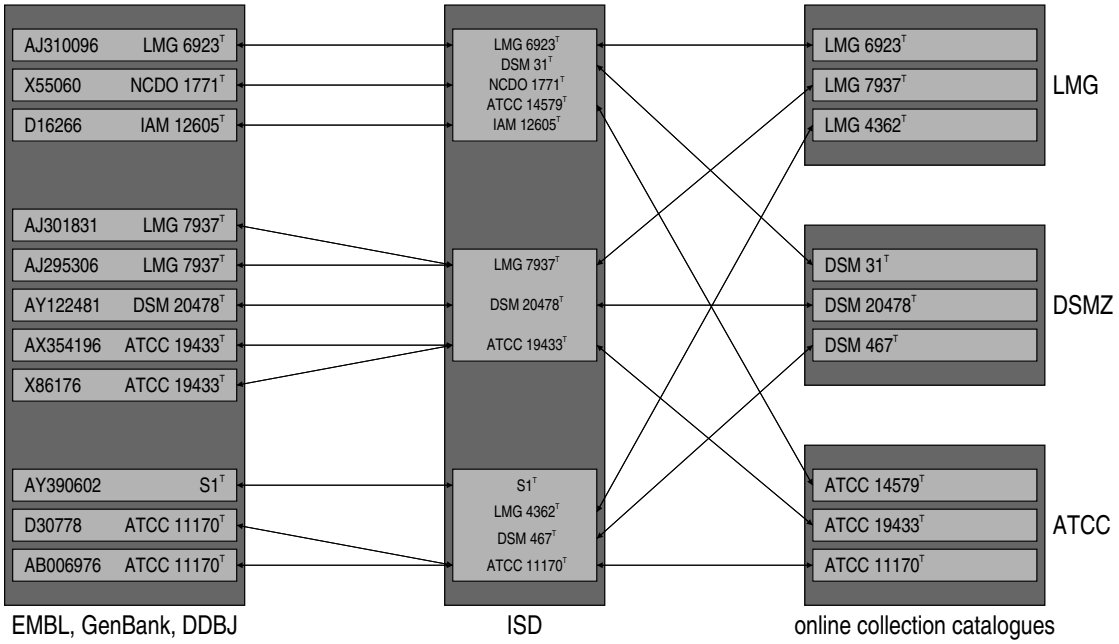


Figure 2.8: Indirect cross-referencing between biological resource centers and peripheral information sources by using an intermediate integrated strain database allows autonomous maintainance of two one-to-many relationships.

only finds the connections indicated by the solid lines. This means that only eight out of eleven sequence records can be associated with a record in the online culture collection catalogues for the simple example show in Figure 2.7. However, the synonymy of the labels dictates that many more indirect links can be found between the sequence records and the culture collection catalogues, as is indicated by the dashed lines in the figure. In order to discover all these indirect links, the complete knowledge of all synonym labels for a given strain must be readily available, whereas at the same time some manual resolution of the ambiguous labels cannot be avoided. And still there is no solution for the lack of unique identifiers for the biological samples in order to instantiate these links in an unambiguous way. Moreover, the management of direct or indirect cross-references between biological resource center records and EMBL database records leads to the cumbersome requirement of maintaining a many-to-many relationship. *Id est*, for each new sequence deposited into the EMBL database, multiple links to the corresponding BRC catalog records may have to be established, and reversely, for each culture deposited into a BRC, multiple references to EMBL records may need to be created. This would truely be an example of bad data management.

Many generic design patterns for software and data organization are founded on the general principle of redirection using one or more intermediate levels, in order to improve the overall flexibility and manageability of the system [29]. In the specific case of linking EMBL records with BRC records, and vice versa, redirection can easily be implemented by making use of the virtues of the integrated strain database, in a way that is illustrated in Figure 2.8. In this flexible cross-referencing scheme, the EMBL records and BRC records are no longer directly linked. Instead, each BRC record is directly linked to the integrated

strain database. Under normal conditions, these reference can be automatically established as the BRCs make use of unique strain labels in most of the case. Similarly, each EMBL record for which a strain label is found can also be direcly linked to the integrated strain database, either automatically if the strain label is unique as defined by the integrated strain database, or manually when it concerns ambiguous labels. In the latter case, the integrated strain database provides the necessary context for resolving the ambiguity of the labels. All cross-references can now be made persistent by using the culture identifiers as provided by the integrated strain database as the unique object identifiers for the biological samples. Note that each record added to either the EMBL database or to a BRC database only requires the establishment of a single reference to the integrated strain database in the scheme of Figure 2.8. Linking a new record from a *peripheral* data source (the adjective peripheral is used for all data sources other than the integrated strain database) may result in the creation of a new culture equivalence class into the integrated strain database, if it concerns a culture that was not known by the integrated strain database, or even a new strain equivalence class if it concerns a newly isolated strain. As such, lookup of the relevant culture identifier in the integrated strain database follows the same general outline of the `IncrementEquivalence` procedure, using a record with only one label as the input parameter. Remark that the cross-referencing scheme of Figure 2.8 resolves all direct and indirect links between peripheral data sources and the integrated strain database using only 20 direct connections with the integrated strain database, which amounts to the total sum of the records in the peripheral data sources, while the cross-referencing scheme of Figure 2.7 needs 33 links between the peripheral data sources in order to accomplish the same set of mutual links. This discrepancy is even more pronounced in real-world situations, where it is envisioned that the BRC records are linked with multiple peripheral data sources, other than the International Nucleotide Sequence Database.

Linkage of peripheral information records with ambiguous strain labels will not result in an automatic resolution of the culture identifier, given the exit command on line 22 of the `IncrementEquivalence` procedure. In these circumstances, human intervention is required in order to sort out the exact semantics of the ambiguous label. To illustrate how this procedure of manual linkage may be set up, let us consider the EMBL sequence records with accession numbers `AF509820`, `AJ309324` and `AJ278726` that all refer to their sequenced bacterial strain using the label ambiguous `B2`. From Table 2.2 we have learned that multiple strains are referenced by this label, so that the label indeed deserves to be tagged as ambiguous. If a closer look is taken to the `AF509820` sequence record, one may find out that the biological sample from which the DNA was extracted is identified as *Acinetobacter baylyi*. This suggests that the EMBL record should be linked to the culture in the integrated strain database entry with identifier 368362, according to the search results presented in Table 2.2. This solution is confirmed by manually looking up the synonym labels in the publication by Carr *et al.* [10] that is linked to EMBL entry `AF509820`. Similarly, the EMBL sequence record with accession number `AJ309324` is identified as *Chryseobacterium defluvii*, indicating that it should be linked to the culture in the integrated strain database with identifier 65830. Again, this is confirmed by the synonym strain labels mentioned in the paper by Kämpfer *et al.* [43] that is linked to this EMBL entry. For the final EMBL record with accession number `AJ278726`, both the identification as the species *Methylocapsa acidiphila* and the synonym labels encoutered in the paper by

Dedysh *et al.* [17] prove that the EMBL record should be linked to the integrated strain database record with culture identifier 60975. Note that in this latter case, some of the synonym labels are also incorporated into the EMBL record itself. After cross-referencing the sequence database with the integrated strain database it can be easily derived that all three of the above EMBL records represent sequences of bacterial type strains, notwithstanding the fact that this information was not directly provided in the EMBL records with accession number `AJ309324`.

Given the manual exertions required to resolve ambiguous strain labels during lookup of the corresponding culture identifier within the integrated strain database, it took us quite some effort to link a selection of 130671 bacterial sequence records that are potentially related to the 16S rRNA gene with the integrated strain database. At present, the attained succes rate of this operation was that only 13636 (10.4%) of these sequence records have been succesfully linked in the way described above. Although a vast number of the currently unlinked records concern sequences related to uncultured or uncultureable bacterial strains, our experience from working with the EMBL sequence database is that still a significant number of the unlinked records can be manually linked, at the cost of a time-consuming lookup process. This may include looking up information within the integrated strain database or within external data sources (mainly publications) linked to the public sequence database. The cross-referencing scenario could be further improved if the information provided by integrated strain database was made publicly accessible as web service, so that culture identifiers could be looked up during the sequence deposit procedure. As a result, the sequence database would be continuously enriched as an effort of the whole research community and advanced searches involving information extracted via the cross-reference links with the biological resource centers would become more reliable. This latter issue is dealt with in subsection 2.5.2.

After establishing solid cross-reference links between the integrated strain database and a peripheral data source such as the public sequence database, it again becomes possible to perform integrity checks on the duplicated information provided by both data providers, in a way much similar as during the construction of the integrated strain database itself. Table 2.10 enumerates a small excerpt of the incorrect strain label references that have been discovered within the International Nucleotide Sequence Database. The first part of this table shows some inconsistencies that were detected by comparison of the strain identification information taken from both the sequence database and the integrated strain database. The first column of this table gives the accession number of the EMBL sequence record together with the associated strain identification stored in the `organism species` field. The second column shows the strain label extracted from the EMBL sequence record with the corresponding strain identification retrieved via the link with the integrated strain database. Clearly, for these examples there is a discrepancy between this identification information that is taken from both data sources. By consulting the literature references linked to the EMBL records and searching the integrated strain database, the cross-reference links have been corrected as indicated in the last column of the table. The last few rows of Table 2.10 show examples of records in the public sequence database that refer to strain labels that do not occur within the online catalogue of the corresponding biological resource center. These incorrect references have been resolved in a similar way

accession number	incorrect strain reference	correct strain reference
M58730 (*Bifidobacterium asteroides*)	ATCC 29510 (*Stenotrophomonas maltophilia*)	ATCC 25910^{T} (*B. asteroides*)
X71855 (*Clostridium xylanolyticum*)	ATCC 4963 (*Lactobacillus gasseri*)	ATCC 49623^{T} (*C. xylanolyticum*)
AB089482 (*Derxia gummosa*)	ATCC 15594 (*Arthrobotrys conoides*)	ATCC 15994^{T} (*D. gummosa*)
Y17361 (*Lactobacillus amylolyticus*)	DSM 1664 (*Clostridium sporogenes*)	DSM 11664^{T} (*L. amylolyticus*)
AB119197 (*Beijerinckia indica*)	ATCC 9036 (*Acinetobacter johnsonii*)	ATCC 9039^{T} (*B. ind.* subsp. *indica*)
X16895 (*Listonella anguillarum*)	ATCC 12964 (*Streptococcus pyogenes*)	ATCC 19264^{T} (*L. anguillarum*)
X80180 (*Acetomicrobium flavidum*)	DSM 20663 (*Lactobacillus sanfranciscensis*)	DSM 20664^{T} (*A. flavidum*)
AJ224308 (*Aeromonas popoffii*)	LMG 317541 (does not exist)	LMG 17541^{T} (*A. popoffii*)
X81623 (*Shewanella putrefaciens*)	LMG 26268 (does not exist)	LMG 2268^{T} (*S. putrefaciens*)
AY655733 (*Corynebacterium sphenisci*)	DSMZ 44792 (does not exist)	DSM 44792^{T} (*C. sphenisci*)
AY543023 (*Brochothrix thermosphacta*)	DSMZ 20171 (does not exist)	DSM 20171^{T} (*B. thermosphacta*)
X55060 (*Bacillus cereus*)	1771 (NCD0) (does not exist)	NCDO 1771^{T} (*B. cereus*)
X74702 (*Vibrio fischeri*)	ATCC 774 (does not exist)	ATCC 7744^{T} (*V. fischeri*)

Table 2.10: Examples of inconsistencies found during cross-referencing the International Nucleotide Sequence Database with the integrated strain database.

as described before.

2.5.2 Advanced dynamic queries

Cross-referencing multiple autonomous and heterogeneous data sources by means of the intermediate level provided by the integrated strain database, in essence creates a transparant information gateway wherein the biological material takes up a centralized role. This allows the execution of all sorts of advanced queries on the fly, which are in part resolved by making use of the knowledge provided by the integrated strain database in combination with information extracted from the peripheral data sources. As a result, such dynamic queries can automatically bridge over multiple data sources that were physically separated before the integration process. Before the establishment of the integrated strain database, answering these advanced queries could be quite a labour-intensive and time-consuming task. Consequently, the search results did not always gave a complete and up-to-date view on the information provided by the in-house and public domain peripheral data sources. Throughout this chapter we already illustrated the applicability of advanced queries for the implementation of some extensive integrity checks on duplicated information that is distributed over several peripheral microbial data sources that are now interconnected through the integrated strain database. In this section we focus on some examples of advanced queries that are of a more direct use for microbiologists.

Knowledge of all the experimental data that is determined for a given microbial strain is primordial for easily setting up large polyphasic databases [71]. As an example, Table 2.11 presents a shortlist of the results from a query that searches within the emperical databases connected to the integrated strain database for all known experimental data generated for the *Enterococcus faecium* type strain. To answer this question, it was required to initially lookup all strain labels that are used for referencing the *E. faecium* type strain and subsequently gather all experimental data linked to each of these strain labels. After assembling an overview of the experimental knowledge, more detailed information may be acquired for each separate experiment, by following the link to the particular peripheral data source. This is the kind of advanced queries that is applied for embellishing the online catalogue

experiment type	experiment date	label	accession number
CHARACTER\API\RAPID ID 32 STREP (2.0)	1998-12-22	LMG 11423	
CHARACTER\API\RAPID ID 32 STREP (2.0)	2003-05-14 17:16:55	LMG 11423	
CHARACTER\FAME\TSBA50 (5.00)	1990-10-25 07:12:50	LMG 8149	
CHARACTER\FAME\TSBA50 (5.00)	1990-10-31 12:51:14	LMG 8149	
CHARACTER\FAME\TSBA50 (5.00)	1992-10-19 17:21:04	LMG 11423	
CHARACTER\FAME\TSBA50 (5.00)	1992-10-19 19:21:39	LMG 12692 t2	
CHARACTER\FAME\TSBA50 (5.00)	1992-10-20 07:13:47	LMG 12692 t1	
FINGERPRINT\REP-PCR (GTG5)	2003-11-19 17:11:53	LMG 11423	
FINGERPRINT\SDS-PAGE		LMG 8149	
FINGERPRINT\SDS-PAGE		LMG 8149 t1	
FINGERPRINT\SDS-PAGE		LMG 8149 t2	
FINGERPRINT\SDS-PAGE		LMG 11423	
FINGERPRINT\SDS-PAGE		LMG 12692 t1	
FINGERPRINT\SDS-PAGE		LMG 12692 t2	
FINGERPRINT\SDS-PAGE		LMG 12692 QC 10/92	
SEQUENCE\DNA\16S rRNA	1994-06-08	JCM 5804	D31676
SEQUENCE\DNA\16S rRNA	1998-01-08	CCUG 542	Y12906
SEQUENCE\DNA\16S rRNA	1998-03-24	JCM 5804	AB012213
SEQUENCE\DNA\16S rRNA	1999-07-22	NCFB 942	Y18294
SEQUENCE\DNA\16S rRNA	2000-07-08	DSM 20477	AJ276355
SEQUENCE\DNA\16S rRNA	2000-11-24	LMG 11423	AJ301830
SEQUENCE\DNA\16S rRNA	2001-12-21	CECT 410	AJ420800
SEQUENCE\DNA\16S-23S rRNA spacer	1997-02-21	ATCC 19434	X87180

Table 2.11: Polyphasic search results showing all experimental data generated for the *Enterococcus faecium* type strain, known within the integrated microbial information gateway.

records of the BCCM™/LMG Bacteria Collection (see Table 2.6) with pointers to all related EMBL sequence records and PubMed literature references. As an example, we refer to the catalog record of the *E. faecium* type strain with the following URL

`http://www.belspo.be/bccm/db/bacteria_details2.asp?num=11423`

The International Nucleotide Sequence Database lacks some of the necessary biological strain information for resolving a direct search to find all 16S rRNA gene sequences of all *Enterococcus* spp. type strains that have been deposited within the public domain repository. The term direct search, used in this particular context, refers to the fact that the search operation only makes use of the information incorporated into the sequence database itself. The only appropriate search option within the public sequence database is to retrieve all 16S rRNA sequences associated with *Enterococcus* strains, followed by manually filtering out the entries that are associated with a type strain. Apart from being very time-consuming, the precision of this search strategy might be affected as well by the presence of outdated identification information and missing, ambiguous or incorrect strain information in the public sequence database. Based on the established cross-reference links between the EMBL sequence records and the integrated strain database, such kinds of queries can be resolved more accurately, resulting in a list of relevant EMBL records for the example query that is shown in Table 2.12. Four consecutive steps were taken by the integrated strain database in order to answer the query. First of all, all validly described species of the genus *Enterococcus* were looked up in the database. The integrated strain database then determined the type strain for each of these taxa, followed by a search for all synonym labels known for each of the type strains. In a last step, all EMBL records associated to any of the selected strain labels were extracted by scanning the fixed cross-reference links

acc nr.	species name	strain number	deposit date	description	size
Y11621	*E. asini*	AS2^{T} (LMG 18727^{T})	1998-06-02	E.asini 16S rRNA gene	1551
D31674	*E. avium*	NCDO 2369^{T} (LMG 10744^{T})	1994-06-15	E.avium gene for 16S ribosomal RNA, partial sequence	166
Y18274	*E. avium*	NCFB 2369^{T} (LMG 10744^{T})	1999-07-22	E.avium (strain NCFB 2369T) 16S rRNA gene	1429
AJ301825	*E. avium*	LMG 10744^{T}	2000-11-24	E.avium 16S rRNA gene, strain LMG 10744	1833
Y12907	*E. avium*	ATCC 14025^{T} (LMG 10744^{T})	1998-01-08	E.avium 16S rRNA gene, partial (strain ATTC 14025…	366
AF133535	*E. avium*	CIP 103019^{T} (LMG 10744^{T})	1999-06-01	E.avium 16S ribosomal RNA gene, partial sequence	1524
X76177	*E. canis*	LMG 12316^{T}	1994-07-30	E.sp. (LMG12316) 16S rRNA gene	1440
AF039903	*E. casseliflavus*	ATCC 25788^{T} (LMG 10745^{T})	1998-02-03	E.casseliflavus 16S ribosomal RNA gene, partial sequence	1509
Y18161	*E. casseliflavus*	NCIMB 11449^{T} (LMG 10745^{T})	1999-07-22	E.casseliflavus 16S rRNA gene, strain NCIMB 11449	1421
AJ301826	*E. casseliflavus*	LMG 10745^{T} (LMG 10745^{T})	2000-11-24	E.casseliflavus 16S rRNA gene, strain LMG 10745	1904
Y12908	*E. casseliflavus*	ATCC 25788^{T} (LMG 10745^{T})	1998-01-08	E.casseliflavus 16S rRNA gene, partial (strain ATTC 25788…	366
AJ420804	*E. casseliflavus*	CECT 969^{T} (LMG 10745^{T})	2001-12-21	E.casseliflavus 16S rRNA gene, strain CECT969T	1451
Y18355	*E. cecorum*	NCDO 2674^{T} (LMG 11741^{T})	1999-07-22	E.cecorum 16S rRNA gene	1409
AJ301827	*E. cecorum*	LMG 12902^{T}	2000-11-24	E.cecorum 16S rRNA gene, strain LMG 12902	1667
AF061009	*E. cecorum*	ATCC 43198^{T} (LMG 11741^{T})	1999-02-08	E.cecorum 16S ribosomal RNA gene, partial sequence	1509
Y12917	*E. cecorum*	CCUG 27299^{T} (LMG 11741^{T})	1998-01-08	E.cecorum 16S rRNA gene, partial (strain CCUG 27299)	366
Y18275	*E. columbae*	NCIMB 13013^{T} (LMG 11740^{T})	1999-07-22	E.columbae (strain NCIMB 13013T)16S rRNA gene	1443
X56422	*E. columbae*	NCIMB 13013^{T} (LMG 11740^{T})	1992-03-12	E.columbae 16S rRNA gene	1493
AJ301828	*E. columbae*	LMG 11740^{T}	2000-11-24	E.columbae 16S rRNA gene, strain LMG 11740	1818
AF061006	*E. columbae*	ATCC 51263^{T} (LMG 11740^{T})	1999-02-08	E.columbae 16S ribosomal RNA gene, partial sequence	1481
Y12918	*E. columbae*	CCUG 27894^{T} (LMG 11740^{T})	1998-01-08	E.columbae 16S rRNA gene, partial (strain CCUG 27894)	366
Y18358	*E. dispar*	NCIMB 13000^{T} (LMG 13521^{T})	1999-07-22	E.dispar 16S rRNA gene	1397
AJ301829	*E. dispar*	LMG 13521^{T}	2000-11-24	E.dispar 16S rRNA gene, strain LMG 13521	1875
AF061007	*E. dispar*	ATCC 51266^{T} (LMG 13521^{T})	1999-02-08	E.dispar 16S ribosomal RNA gene, partial sequence	1514
Y12920	*E. dispar*	CCUG 33309^{T} (LMG 13521^{T})	1998-01-08	E.dispar 16S rRNA gene, partial (strain CCUG 33309)	366
Y18359	*E. durans*	NCFB 596^{T} (LMG 10746^{T})	1999-07-22	E.durans 16S rRNA gene	1434
AJ276354	*E. durans*	DSM 20633^{T} (LMG 10746^{T})	2000-07-08	E.durans 16S rRNA gene, strain DSM20633	1534
Y12909	*E. durans*	ATCC 19432^{T} (LMG 10746^{T})	1998-01-08	E.durans 16S rRNA gene, partial (strain ATTC 19432…	366
AJ420801	*E. durans*	CECT 411^{T} (LMG 10746^{T})	2001-12-21	E.durans 16S rRNA gene, strain CECT411T	1506
D31675	*E. faecalis*	NCDO 581^{T} (LMG 7937^{T})	1994-06-08	E.faecalis gene for 16S ribosomal RNA, partial sequence	199
Y18293	*E. faecalis*	NCIMB 775^{T} (LMG 7937^{T})	1999-07-22	E.faecalis 16S rRNA gene	1449
AJ301831	*E. faecalis*	LMG 7937^{T}	2000-11-24	E.faecalis 16S rRNA gene, strain LMG 7937	1556
AB012212	*E. faecalis*	JCM 5803^{T} (LMG 7937^{T})	1998-03-24	E.faecalis gene for 16S rRNA, partial sequence	1517
L16515	*E. faecalis*	NCTC 775^{T} (LMG 7937^{T})	1993-05-20	E.faecalis (NCTC 775) 16S ribosomal RNA,…	418
Y12905	*E. faecalis*	ATCC 19433^{T} (LMG 7937^{T})	1998-01-08	E.faecalis 16S rRNA gene, partial (strain …	366
AJ420803	*E. faecalis*	CECT 481^{T} (LMG 7937^{T})	2001-12-21	E.faecalis 16S rRNA gene, strain CECT481T	1477
D31676	*E. faecium*	JCM 5804^{T} (LMG 11423^{T})	1994-06-08	E.faecium gene for 16S ribosomal RNA, partial sequence	179
Y18294	*E. faecium*	NCFB 942^{T} (LMG 11423^{T})	1999-07-22	E.faecium 16S rRNA gene	1459
AJ276355	*E. faecium*	DSM 20477^{T} (LMG 11423^{T})	2000-07-08	E.faecium 16S rRNA gene, strain DSM20477	1533
AJ301830	*E. faecium*	LMG 11423^{T}	2000-11-24	E.faecium 16S rRNA gene, strain LMG 11423	1651
AB012213	*E. faecium*	JCM 5804^{T} (LMG 11423^{T})	1998-03-24	E.faecium gene for 16S rRNA, partial sequence	1523
Y12906	*E. faecium*	ATCC 19434^{T} (LMG 11423^{T})	1998-01-08	E.faecium 16S rRNA gene, partial (strain ATTC 19434…	366
AJ420800	*E. faecium*	CECT 410^{T} (LMG 11423^{T})	2001-12-21	E.faecium 16S rRNA gene, strain CECT410T	1489
Y18295	*E. flavescens*	NCIMB 13226^{T} (LMG 13518^{T})	1999-07-22	E.flavescens 16S rRNA gene	1425
AJ301832	*E. flavescens*	LMG 13518^{T}	2000-11-24	E.flavescens 16S rRNA gene, strain LMG 13518	1847
Y12923	*E. flavescens*	CCUG 30567^{T} (LMG 13518^{T})	1998-01-08	E.flavescens 16S rRNA gene, partial (strain CCUG 30567)	366
AJ420802	*E. flavescens*	CECT 4481^{T} (LMG 13518^{T})	2001-12-21	E.flavescens 16S rRNA gene, strain CECT4481T	1514
AJ301833	*E. gallinarum*	LMG 13129^{T}	2000-11-24	E.gallinarum 16S rRNA gene, strain LMG 13129	1568
Y12910	*E. gallinarum*	CCUG 18658^{T} (LMG 11207^{T})	1998-01-08	E.gallinarum 16S rRNA gene, partial (strain CCUG 18658)	366
AF039900	*E. gallinarum*	ATCC 49573^{T} (LMG 11207^{T})	1998-02-03	E.gallinarum 16S ribosomal RNA gene, partial sequence	1506
AJ420805	*E. gallinarum*	CECT 970^{T} (LMG 11207^{T})	2001-12-21	E.gallinarum 16S rRNA gene, strain CECT970T	1516
AY033814	*E. gilvus*	PQ1^{T} (CCUG 45553^{T})	2002-04-04	E.gilvus 16S ribosomal RNA gene, partial sequence	1295
AF286832	*E. haemoperoxidus*	CCM 4851^{T} (LMG 19487^{T})	2001-07-11	E.haemoperoxidus 16S ribosomal RNA gene, partial sequence	1512
Y18354	*E. hirae*	NCFB 1258^{T} (LMG 6399^{T})	1999-07-22	E.hirae 16S rRNA gene	1445
Y17302	*E. hirae*	DSM 20160^{T} (LMG 6399^{T})	1999-02-24	E.hirae 16S rRNA gene	1535
AJ276356	*E. hirae*	DSM 20160^{T} (LMG 6399^{T})	2000-07-08	E.hirae 16S rRNA gene, strain DSM20160	1535
AJ301834	*E. hirae*	LMG 6399^{T}	2000-11-24	E.hirae 16S rRNA gene, strain LMG 6399	1787
AF061011	*E. hirae*	ATCC 8043^{T} (LMG 6399^{T})	1999-02-08	E.hirae 16S ribosomal RNA gene, partial sequence	1507
Y12912	*E. hirae*	ATCC 8043^{T} (LMG 6399^{T})	1998-01-08	E.hirae 16S rRNA gene, partial (strain ATTC 8043…	366
AJ420799	*E. hirae*	CECT 279^{T} (LMG 6399^{T})	2001-12-21	E.hirae 16S rRNA gene, strain CECT279T	1514
Y18339	*E. malodoratus*	NCFB 846^{T} (LMG 10747^{T})	1999-07-22	E.malodoratus 16S rRNA gene	1461
AJ301835	*E. malodoratus*	LMG 10747^{T} (LMG 10747^{T})	2000-11-24	E.malodoratus 16S rRNA gene, strain LMG 10747	1701
AF061012	*E. malodoratus*	ATCC 43197^{T} (LMG 10747^{T})	1999-02-08	E.malodoratus 16S ribosomal RNA gene, partial sequence	1500
Y12911	*E. malodoratus*	CCUG 30572^{T} (LMG 10747^{T})	1998-01-08	E.malodoratus 16S rRNA gene, partial (strain …	366
AF286831	*E. moraviensis*	CCM 4856^{T} (LMG 19486^{T})	2001-07-11	E.moraviensis 16S ribosomal RNA gene, partial sequence	1509
Y18340	*E. mundtii*	NCFB 2375^{T} (LMG 10748^{T})	1999-07-22	E.mundtii 16S rRNA gene	1447
AJ301836	*E. mundtii*	LMG 10748^{T}	2000-11-24	E.mundtii 16S rRNA gene, strain LMG 10748	1864
AF061013	*E. mundtii*	ATCC 43186^{T} (LMG 10748^{T})	1999-02-08	E.mundtii 16S ribosomal RNA gene, partial sequence	1529
Y12913	*E. mundtii*	CCUG 18656^{T} (LMG 10748^{T})	1998-01-08	E.mundtii 16S rRNA gene, partial (strain CCUG 18656)	366
AJ420806	*E. mundtii*	CECT 972^{T} (LMG 10748^{T})	2001-12-21	E.mundtii 16S rRNA gene, strain CECT972T	1521
AY033815	*E. pallens*	PQ2^{T} (CCUG 45554^{T})	2002-04-04	E.pallens 16S ribosomal RNA gene, partial sequence	1294
AY028437	*E. phoeniculicola*	JLB-1^{T} (DSM 14726^{T})	2001-07-02	E.phoeniculicola 16S ribosomal RNA gene,…	1479
Y18356	*E. pseudoavium*	NCFB 2138^{T} (LMG 11426^{T})	1999-07-22	E.pseudoavium 16S rRNA gene	1424
AJ301837	*E. pseudoavium*	LMG 11426^{T}	2000-11-24	E.pseudoavium 16S rRNA gene, strain LMG 11426	1636
AF061002	*E. pseudoavium*	ATCC 49372^{T} (LMG 11426^{T})	1999-02-08	E.pseudoavium 16S ribosomal RNA gene, partial sequence	1513
Y12916	*E. pseudoavium*	CCUG 33310^{T} (LMG 11426^{T})	1998-01-08	E.pseudoavium 16S rRNA gene, partial (strain…	366
Y18296	*E. raffinosus*	NCIMB 12901^{T} (LMG 12888^{T})	1999-07-22	E.raffinosus 16S rRNA gene	1425
Y12914	*E. raffinosus*	CCUG 29292^{T} (LMG 12888^{T})	1998-01-08	E.raffinosus 16S rRNA gene, partial (strain …	366
AF326472	*E. ratti*	DS 2705-87^{T} (NCIMB 13635^{T})	2002-11-21	E.ratti 16S ribosomal RNA gene, partial sequence	1523
AF539705	*E. ratti*	ATCC 700914^{T} (NCIMB 13635^{T})	2002-09-12	E.ratti 16S ribosomal RNA gene, partial sequence	1503
X55766	*E. saccharolyticus*	NCDO 2594^{T} (LMG 11427^{T})	1992-03-12	S.saccharolyticus 16S rRNA gene (5')	144
Y18357	*E. saccharolyticus*	NCDO 2594^{T} (LMG 11427^{T})	1999-07-22	E.saccharolyticus 16S rRNA gene	1456
AJ301839	*E. saccharolyticus*	LMG 11427^{T}	2000-11-24	E.saccharolyticus 16S rRNA gene, strain LMG 11427	1902
U30931	*E. saccharolyticus*	NCDO 2594^{T} (LMG 11427^{T})	1996-07-31	E.saccharolyticus 16S ribosomal RNA partial sequence	1521
AF061004	*E. saccharolyticus*	ATCC 43076^{T} (LMG 11427^{T})	1999-02-08	E.saccharolyticus 16S ribosomal RNA gene, partial sequence	1506
Y12919	*E. saccharolyticus*	ATCC 43076^{T} (LMG 11427^{T})	1998-01-08	E.saccharolyticus 16S rRNA gene, partial (strain ATTC 43076…	366
X55767	*E. saccharolyticus*	NCDO 2594^{T} (LMG 11427^{T})	1992-03-12	S.saccharolyticus 16S rRNA gene	1293
Y18338	*E. solitarius*	NCIMB 12902^{T} (LMG 12890^{T})	1999-07-22	E.solitarius 16S rRNA gene	1411
AF061010	*E. solitarius*	ATCC 49428^{T} (LMG 12890^{T})	1999-02-08	E.solitarius 16S ribosomal RNA gene,…	1341
AJ301840	*E. solitarius*	DSM 5634^{T} (LMG 12890^{T})	2000-11-24	E.solitarius 16S rRNA gene, strain DSM 5634	1653
Y12915	*E. solitarius*	CCUG 29293^{T} (LMG 12890^{T})	1998-01-08	E.solitarius 16S rRNA gene, partial (strain CCUG 29293)	367
Y18341	*E. sulfureus*	NCIMB 13117^{T} (LMG 13084^{T})	1999-07-22	E.sulfureus 16S rRNA gene	1391
X55133	*E. sulfureus*	MUTK 31^{T} (LMG 13084^{T})	1991-06-17	E.sulfureus 16S ribosomal RNA	1495
AJ301841	*E. sulfureus*	LMG 13084^{T}	2000-11-24	E.sulfureus 16S rRNA gene, strain LMG 13084	1902
AF061001	*E. sulfureus*	ATCC 49903^{T} (LMG 13084^{T})	1999-02-08	E.sulfureus 16S ribosomal RNA gene, partial sequence	1498
Y12921	*E. sulfureus*	CCUG 33313^{T} (LMG 13084^{T})	1998-01-08	E.sulfureus 16S rRNA gene, partial (strain CCUG 33313)	366
AJ271329	*E. villorum*	LMG 12287^{T}	2001-06-13	E.villorum 16S rRNA gene, strain LMG 12287	1512

Table 2.12: Integrated microbial information gateway search results showing 16S rRNA gene sequences of all *Enterococcus* spp. type strains, deposited within the International Nucleotide Sequence Database.

between the integrated strain database and the public nucleotide sequence database. By application of the dynamic search approach based on the information collected in and the cross-references build around the integrated strain database, a total of 97 relevant records were found for the example query. As a reference, the most recent version of the *Taxonomic Outline of the Prokaryotes* [31] only contains 39 of these 16S rRNA sequence records, after performing some data reduction based on sequence quality evaluation. Some futher remarks can be made on the search results shown in Table 2.12. It is clear that strain labels that are ambiguous or are not related to large culture collections need to be incorporated as well into solid cross-referencing schemes for connecting microbial data sources, such as the one discussed in subsection 2.5.1. Otherwise no 16S rRNA sequence records would have been found for *Enterococcus asini*, *E. gilvus*, *E. pallens* and *E. phoeniculicola* for this particular query. The importance of accurate quality control on the information duplicated into multiple autonomous data sources is illustrated in the EMBL records with accession numbers `Y12906` (*E. faecium*), `Y12907` (*E. avium*), `Y12908` (*E. casseliflavus*), `Y12909` (*E. durans*), `Y12912` (*E. hirae*) and `Y12916` (*E. pseudoavium*), where the acronym of the strain label has been misspelled as `ATTC`, instead of using the correct acronym `ATCC` of the American Type Culture Collection. Such anomalies are easily detected and corrected during the establishment of cross-reference links with integrated strain database. We also note that the EMBL record with accession number `X76177` will never be retrieved from queries that directly search for *E. canis* sequences, given the inaccurate identification information encoded into the EBML database for this record. For reasons of completeness, we finally remark that *E. porcinus* was found to be a junior synonym of *E. villorum* [18], while *E. seriolicida* was reclassified as *Lactococcus garvieae* [25]. The sequence records linked to the type strains of these heterotypic synonyms have been discarded from the search results shown in Table 2.12.

2.6 Conclusions and future perspectives

The present chapter has sketched the need for establishing a divide and conquer strategy for the management of distributed microbial information providers, wherein a logical central repository could take up the responsability to provide a concise, complete and correct view on the semantic equivalences of the labels assigned to microbial strains and cultures. With the evocation of an integrated strain database for the accumulation of synonym label equivalences gathered from a battery of data sources, we have made an initial attempt to implement such a central repository. Our goal to achieve both completeness and correctness of the information content of the integrated strain database, requires a perpetual engagement in processing new and updated data sources, while monitoring the quality of the incorporated data. In this respect, not only plans are at stake to process the information of additional bacterial data sources, but also to widen the scope to other kinds of microorganisms such as fungi and yeasts. We have also indirectly put forward how the integrated strain database may introduce a system of unique indentifiers for resolving homonyny within the currently applied microbial labelling mechanism and simultaneously provide the necessary contextual information to settle the occurrences of ambiguous labels in peripheral data sources. A web-enabled interface to the strain and culture equivalence relations could

thus help to semi-automatically build cross-reference links between the integrated strain database and data sources that contain related strain information. As such, the integrated strain database might serve as a basic building block within an information gateway that seemlessly glues together all related pieces of the taxonomic puzzle. This could feed the application of a multitude of data mining techniques for the discovery of valuable new insights within the data. We equally foresee a role for the automatic integration of complete strain history information within the integrated strain database. This tracking and tracing of the dissemination of microbial strains might as such give support to some quality control and intellectual property right issues. Despite this slew of unresolved issues, the authors hope that the ideas behind the integrated strain database might finally lead to some global action in the integration of microbial data sources, instead of just wishful thinking.

Bibliography

[1] Access to Biological Collection Data (ABCD). `http://www.bgbm.org/TDWG/CODATA`.

[2] **Ace, J., Marvel, B. & Richer, B. (1992).** Matchmaker...matchmaker...find me the address (exact address match processing). *Telephone Engineer and Management* **96(8)**, 50–53.

[3] **Anderberg, M. R. (1973).** Cluster analysis for applications. *Academic Press*, New York and London.

[4] **Batini, C., Lenzerini, M., & Navathe, S. (1986).** A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys* **18(4)**, 323–364.

[5] **Bitton, D. & DeWitt, D. J. (1983).** Duplicate record elimination in large data files. *ACM Transactions of Database Systems* **8(2)**, 255–265.

[6] **Boyer, R. S. & Strother Moore, J. (1977).** A fast string–searching algorithm. *Communications of the ACM* **20(10)**, 762–772.

[7] **Broder, A., Glassman, S., Manasse, M. & Zweig, G. (1997).** Syntactic clustering of the web. In: *Proceedings of the Sixth International World Wide Web Conference*, 391–404.

[8] **Buneman, P., Davidson, S., Hart, K., Overton, C. & Wong, L. (1995).** A data transformation system for biological data sources. In: *Proceedings of the 21st Very Large Data Base (VLDB) Conference*, Zurich, Switzerland, 1995.

[9] CABRI (1998). Guideline for Catalogue Production. `www.cabri.org`.

[10] **Carr, E. L., Kämpfer, P., Patel, B. K. C., Gürtler, V. & Seviour, R. J. (2003).** Seven novel species of *Acinetobacter* isolated from activated sludge. *Int J Syst Evol Microbiol* **53**, 953–963.

[11] **Cattell, R. & Barry, D. (1997).** The Object Database Standard: ODMG 2.0. Morgan Kaufmann Publishers.

[12] **Chang, W. I. & Lampe, J. (1992).** Theoretical and empirical comparisons of approximate string matching algorithms. In: *3rd Symposium on Combinatorial Pattern Matching* (CMP), 175–184.

[13] **Cohen, W. W., Kautz, H. A. & McAllester, D. A. (2000).** Hardening soft information sources. In: *Proceedings of the Sixth International Conference on Knowledge Discovery and Data Mining* (KDD-2000), 255–259.

[14] **Cormen, T. H., Lesierson, C. E. & Rivest, R. L. (1990).** Introduction to Algorithms. MIT Press.

[15] The Darwin Core.
`http://tsadev.speciesanalyst.net/DarwinCore/darwin_core.asp`.

[16] **Dawyndt, P., Vancanneyt, M. & Swings, J. (2004).** On the integration of microbial information. *WFCC Newsletter*, **38**, 19–34.

[17] **Dedysh, S. N., Khmelenina, V. N., Suzina, N. E., Trotsenko, Y. A., Semrau, J. D., Liesack, W. and Tiedje, J. M. (2002).** *Methylocapsa acidiphila* gen. nov., sp. nov., a novel methane-oxidizing and dinitrogen-fixing acidophilic bacterium from *Sphagnum* bog. *Int J Syst Evol Microbiol*, **52**, 251–261.

[18] **De Graef, E. M., Devriese, L. A., Vancanneyt, M., Baele, M., Collins, M. D., Lefebvre, K., Swings, J. & Haesebrouck, F. (2003).** Description of *Enterococcus canis* sp. nov. from dogs and reclassification of *Enterococcus porcinus* Teixeira et al. 2001 as a junior synonym of *Enterococcus villorum* Vancanneyt et al. 2001. *Int J Syst Evol Microbiol* **53(4)**, 1069–1074.

[19] **De Meyer, H., Naessens, H. and De Baets, B. (2004).** Algorithms for computing the min-transitive closure and associated partition tree of a symmetric fuzzy relation. *European J Oper Res*, **155**, 226–238.

[20] **Dice, L. R. (1945).** Measures of the amount of ecological association between species. *J Ecology*, **26**, 297–302.

[21] **Dijkshoorn, L., Ursing, B. M. & Ursing, J. B. (2000).** Strain, clone and species: comments on three basic concepts of bacteriology. *J Med Microbiol*, **49(5)**, 397–401.

[22] **Dimitriadou, E., Dolnicar, S. & Weingessel, A. (2002).** An examination of indexes for determining the number of clusters in binary data sets. *Psychometrika* **67(1)**, 137–160.

[23] **Du, M.-W. & Chang, S. C. (1994).** Approach to designing very fast approximate string matching algorithms. *IEEE Transactions on Knowledge and Data Engineering* **6(4)**, 620–633.

[24] **Dunn, J. (1974).** A graph-theoretical analysis of pattern classification via Tamura's fuzzy relation. *IEEE Transactions on Systems, Man and Cybernetics* **4(3)**, 310–313.

[25] **Eldar, A., Ghittino, C., Asanta, L., Bozzetta, E., Goria, M., Prearo, M. & Bercovier, H. (1996).** *Enterococcus seriolicida* is a junior synonym of *Lactococcus garvieae*, a causative agent of septicemia and meningoencephalitis in fish. *Curr Microbiol* **32(2)**, 85–88.

[26] **Fayyad, U., Piatetsky-Shaprio, G. & Smyth, P. (1996).** From data mining to knowledge discovery in databases. *AI Magazine* **17(3)**, 37–54.

[27] **Fellegi, I. P. & Sunter, A. B. (1969).** A theory for record linkage. *Journal of the American Statistical Association* **64**, 1183–1210.

[28] **Galil, Z. & Giancarlo, R. (1988).** Data structures and algorithms for approximate string matching. *Journal of Complexity* **4**, 33–72.

[29] **Gamma, E., Helm, R., Johnson, R. & Vlissides, J. M. (1995).** Design Patterns: Elements of Reusable Object-Oriented Software. Addison Wesley, Reading, MA, USA.

[30] **Gams, W., Hennebert, G. L., Stalpers, J. A., Janssens, D., Schipper, M. A. A., Smith, J., Yarrow, D. & Hawksworth, D. L. (1988).** Structuring strain data for storage and retrieval of information on fungi and yeasts in MINE, the Microbial Information Network Europe. *J Gen Microbiol*, **134**, 1667–1689.

[31] **Garrity, G. M., Bell, J. A. & Lilburn, T. G. (2004).** Taxonomic Outline of the Prokaryotes. In: *Bergey's Manual of Systematic Bacteriology*, Volume 2: The Proteobacteria, 2nd Edition, Release 5.0, Springer–Verlag, New York, NY, USA. `DOI:10.1007/bergeysoutine200405`.

[32] **Gyllenberg, M., Koski T. & Verlaan, M. (1997).** Classification of binary vectors by stochastic complexity. *J Multivariate Anal* **63**, 47–72.

[33] **Halkidi, M. & Vazirgiannis, M. (2001).** Clustering validity assessment: finding the optimal partitioning of a data set. In: *Proceedings of the IEEE International Conference on Data Mining*, California, USA, November 2001.

[34] **Hall, P. A. V. & Dowling, G. R. (1980).** Approximate string matching. *ACM Computing Surveys* **12(4)**, 381–402.

[35] **Hartigan, J. A. (1975).** Clustering algorithms. Willey, New York, NY, USA.

[36] **Hernández, M. A. & Stolfo, S. J. (1995).** The merge\purge problem for large databases. In: *Proceedings of the ACM-SIGMOD International Conference on Management of Data*, May 1995, 127–138.

[37] **Hernández, M. A. & Stolfo, S. J. (1998).** Real–world data is dirty: data cleansing and the merge\purge problem. *Journal of Data Mining and Knowledge Discovery* **2(1)**, 9–37.

[38] **Hopcroft, J. E. & Ullman, J. D. (1973).** Set merging algorithms. *SIAM Journal on Computing* **2(4)**, 294–303.

[39] **Hylton, J. A. (1996).** Identifying and merging related bibliographic records. M.S. thesis, MIT, Published as MIT Laboratory for Computer Science Technical Report 678.

[40] International Nucleotide Sequence Database, publicly accessible through the DDBJ (`www.ddbj.nig.ac.jp/Welcome.html`), EMBL (`www.ebi.ac.uk/embl/index.html`) and GenBank (`www.ncbi.nlm.nih.gov`) portals.

[41] **Jacquemin, C. & Royaute, J. (1994).** Retrieving terms and their variants in a lexicalized unification–based framework. In: *Proceedings of the ACM-SIGIR Conference on Research and Development in Information Retrieval*, 132–141.

[42] **Josifovski, V. & Risch, T. (1999).** Integrating heterogeneous overlapping databases through object-oriented transformations. In: *25th International Conference On Very Large Databases*, Edinburgh, Scotland, September 1999, 435–446.

[43] **Kämpfer, P., Dreyer, U., Neef, A., Dott, W. & Busse, H.-J. (2003).** *Chryseobacterium defluvii* sp. nov., isolated from wastewater. *Int J Syst Evol Microbiol*, **53**, 93–97.

[44] **Kim, W., Choi, I., Gala, S. & Scheevel, M. (1993).** On resolving schematic heterogeneity in multidatabase systems. *Distributed and Parallel Databases* **1(3)**, 251–279.

[45] **Knuth, D. E., Morris, J. H. Jr. & Pratt, V. R. (1977).** Fast pattern matching in strings. *SIAM Journal on Computing* **6(2)**, 323–350.

[46] **Koutsofios, E. & North, S. C. (1994)**. Drawing graphs with dot. `http://research.att.com:dist/drawdag/dotdoc.ps.Z`.

[47] **Kukich, K. (1992).** Techniques for automatically correcting words in text. *ACM Computing Surveys* **24(4)**, 377–439.

[48] **Kundu, S. (2000).** An optimal $\mathcal{O}(N^2)$ algorithm for computing the min-transitive closure of a weighted graph. *Inform Proc Lett* **74(5–6)**, 215–220.

[49] **Lane, P. & Lumpkin, G. (1999).** Oracle8*i* Data Warehousing Guide, Release 2 (8.1.6), Oracle Corporation, USA.

[50] **Levenshtein, V. (1966).** Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics* **10**, 707–710.

[51] **Lee, H. (2001).** An optimal algorithm for computing the max-min transitive closure of a fuzzy similarity matrix. *Fuzzy Sets and Systems* **123(1)**, 129–136.

[52] **Madhavaram, M., Ali, D. L. & Zhou, M. (1996).** Integrating heterogeneous distributed database systems. *Computers & Industrial Engineering* **31(1-2)**, 315–318.

[53] **Mecella, M., Scannapieco, M., Virgillito, A., Baldoni, R., Catarci, T. & Batini, C. (2003).** Managing data quality in cooperative information systems. *Journal of Data Semantics*, Volume 1, LNCS.

[54] **Milo, T. & Zohar, S. (1998).** Using schema matching to simplify heterogeneous data translation. Gupta, A., Shmueli, O. & Widom, J. (eds.). In: *Proceedings of the 24rd International Conference on Very Large Data Bases* (VLDB'98), August 24–27 1998, Morgan Kaufman, New York, NY, USA, 122–133.

[55] **Monge, A. E. & Elkan, C. P. (1996).** The WebFind tool for finding scientific papers over the worldwide web. In: *Proceedings of the 3rd International Congress on Computer Science Research*, Tijuana, Baja California, México, 41–46.

[56] **Monge, A. E. & Elkan, C. P. (1997).** An efficient domain-independent algorithm for detecting approximately duplicate database records. In: *Proceedings of the SIGMOD'97 Workshop on Research Issues on Data Mining and Knowledge Discovery* (DMKD'97), Tucson, AZ, USA, May 1997.

[57] **Monge, A. E. (2000).** An adaptive and efficient algorithm for detecting approximately duplicate database records. Technical Paper 90840-8302, California State University, Long Beach, CECS Department, Long Beach, CA, USA.

[58] **Needleman, S. B. & Wunsch, C. D. (1970).** A general method applicable to the search for similarities in the amino acid sequences of two proteins. *Journal of Molecular Biology* **48**, 443–453.

[59] **Newcombe, H. B., Kennedy, J. M., Axford, S. J. & James, A. P. (1959).** Automatic linkage of vital records. *Science* **130**, 954–959.

[60] **Newcombe, H. B. (1988).** Handbook of record linkage: methods for health and statistical studies, administration and business. Oxford University Press.

[61] **Peterson, J. (1980).** Computer programs for detecting and correcting spelling errors. *Communications of the ACM* **23(12)**, 676–687.

[62] **Ramakrishnan, N. & Grama, A. Y. (2001).** Mining scientific data. *Advances in Computers* **55**, 119–169.

[63] **Preißner, R., Goede, A. & Frömmel, C. (1999).** Homonyms and synonyms in the dictionary of interfaces in proteins (DIP). *Bioinformatics* **15**, 832–836.

[64] **Senator, T. E., Goldberg, H. G., Wooton, J., Cottini, M. A., Umarkhan, A. F., Klinger, C. D., Llamas, W. M., Marrone, M. P. and Wong, R. W. (1995).** The financial crimes enforcement network AI system (FAIS): identifying potential money laundring from reports of large cash transactions. *AI Magazine*, **16(4)**, 21–39.

[65] **Slaven, B. E. (1992).** The set theory matching system: an application to ethnographic research. *Social Science Computer Review*, **10(2)**, 215–229.

[66] **Smith, T. F. & Waterman, M. S. (1981).** Identification of common molecular subsequences. *Journal of Molecular Biology* **147)**, 195–197.

[67] **Sneath, P. H. A. & Sokal, R. R. (1973).** Numerical Taxonomy. The Principles and Practice of Numerical Classification. W. H. Freeman and Co., San Francisco.

[68] **Song, W. W., Johannesson, P. & Bubenko, J. A. Jr. (1996).** Semantic similarity relations and computation in schema integration. *Data & Knowledge Engineering* **19(1)**, 65–97.

[69] **Staley, J. T. & Krieg, N. R. (1984).** Classification of procaryotic organisms: an overview. In: *Bergeys Manual of Systematic Bacteriology*, Williams and Wilkins, Baltimore, MD, USA, 1–4.

[70] **Stalpers, J. A., Kracht, M., Janssens, D., De Ley, J., Van Der Toorn, J., Smith, J., Claus, D. & Hippe, H. (1990).** Structuring strain data for storage and retrieval of information on bacteria in MINE, the Microbial Information Network Europe. *Syst Appl Microbiol* **13**, 92–103.

[71] **Vandamme, P., Pot, B., Gillis, M., De Vos, P., Kersters, K. & Swings, J. (1996).** Polyphasic taxonomy, a consensus approach to bacterial systematics. *Microbiological Review* **60**, 407–438.

[72] **Wang, Y. R., Madnick, S. E. & Horton, D. C. (1989).** Inter–database instance identification in composite information systems. In: *Proceedings of the 22nd Annual Hawaii International Conference on System Sciences*, 1989, 677–684.

[73] **Weiss, N., Kracht, M., Gleim, D. & Tindall, B. J.** Bacterial Nomenclature Up-to-date. DSMZ-Deutsche Sammlung von Mikroorganismen und Zellkulturen GmbH, Braunschweig, Germany.
`http://www.dsmz.de/bactnom/bactname.htm`.

[74] WFCC-MIRCEN, World Data Centre for Microorganisms (WDCM), directory of culture collections. `http://wdcm.nig.ac.jp`.

[75] **Yampolskii, M. I. & Gorbonosov, A. E. (1973).** Detection of duplicate secondary documents. *Nauchno-Tekhnicheskaya Informatsiya*, **1(8)**, 3–6.

Chapter 3

The content of this chapter is a strongly based on the published or submitted material in the following scientific journal papers:

[1] **Dawyndt, P., De Meyer, H., De Baets, B. & Swings, J. (2002).** A fast algorithm for generating a min-transitive opening of a similarity relation. In: *Proceedings of the EUROFUSE Workshop on Information Systems*, Villa Monastero, Varenna, Italy, 2002.

[2] **Dawyndt, P., De Meyer, H. & De Baets, B. (in press).** The complete linkage clustering algorithm revisited. *Soft Computing*. DOI: 10.1007/s00500-003-0346-3.

[3] **Dawyndt, P., De Meyer, H. & De Baets, B. (2004).** On the min-transitive approximation of symmetric fuzzy relations. In: *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2004)*, Budapest, Hungary, 25–29 July, 2004.

Chapter 3

Min-transitive Approximations of Similarity Relations

"There are many meaningful groupings"

— Michael R. Anderberg

MIN-TRANSITIVE similarity relations are in one-to-one correspondence to hierarchical partition trees. These hierarchies somewhat resemble the diversification of prokaryotic life according to the Darwinian theory of evolution [4]. As a result, the application of hierarchical clustering methods for the approximative representation of empirical similarity models into a stratified manner already has a long-term operational tradition for the interpretation of relationships among groups of organisms in bacteriology, ranging from the development of complete taxonomies [20] to the delineation of the different subspecies of a distinct but varied species [41].

A new algorithm for generating a reflexive and symmetric min-transitive opening of a given similarity relation is proposed. Since min-transitive similarity relations are nothing else but hierarchical partition trees, the new opening algorithm can thus be compared to certain classical clustering algorithms. Various tests will illustrate that the new algorithm is efficient and the generated opening is in practical situations usually a more reliable representation of the original similarity relation than are the openings generated by other algorithms. In addition, two new algorithms are proposed for generating a min-transitive approximation of a given similarity relation, which in general deviates less from the given similarity relation than are its min-transitive closure and openings, and which is guaranteed to be still reflexive and symmetric. Since the new algorithms are weight-driven, they can be used to generate layer by layer the partition tree associated to the corresponding min-transitive approximation. We report on numerical tests that have been carried out on synthetic data to compare the approximations generated by the new algorithms to the min-transitive closure,

a representative min-transitive opening and the min-transitive approximation delivered by the UPGMA clustering algorithm.

3.1 Introduction

In numerical taxonomy, and in particular in the classification and identification of microbial organisms, one disposes of a wide variety of mathematical methods for grouping taxonomical units into taxa on the basis of their phenotypic and genotypic characteristics. Certain of these classification methods rely on the optimisation of an information-theoretic expression, such as entropy or stochastic complexity [8, 21, 22]. Other methods such as multi-dimensional scaling (MDS) and principal component analysis (PCA) are based on a reduction of the feature space into a two or three dimensional representation. In this chapter, we focus on another class of methods, the similarity-based hierarchical clustering methods, which make use of an intermediate similarity matrix that is built by means of an appropriate similarity measure. Upon that matrix, different classical clustering techniques can be applied, such as single linkage clustering, complete linkage clustering [36], unweighted pair-group method using arithmetic averages (UPGMA, [35]) clustering, Ward's method [42], neighbour joining [34], . . ., in order to obtain a taxonomic stratification, usually represented in the form of a (hierarchical) partition tree or a dendrogram. Algebraically, such a partition tree is equivalent to a min-transitive similarity matrix. Hence, any of these methods turns out to be a way of deriving a min-transitive similarity matrix from the given similarity matrix. Typically, a single partition of clusters is obtained by cutting the tree at some level, or equivalently, by taking a cut of the corresponding min-transitive similarity matrix.

Where the first group of methods, although based on well-founded theoretical principles, suffers in practice from a considerable time-complexity inherent to the underlying optimisation problem, and the scaling methods only perform well for data sets with a small number of interesting groups, the weakness of the hierarchical clustering methods lies in the overwhelming variety of ad-hoc choices that must be made (e.g. the choice of a similarity measure, a clustering method, an optimal cut off level) and the fact that with every possible choice, in general, a different partition is obtained.

Nowadays, sophisticated software tools are available for the classification of bacteria which cover a wide, representative range of classification methods. The abundance of methods forces one to focus on the problem of identifying additional criteria for selecting a particular classification method and for obtaining an optimal partition of clusters. In practice, the criteria used are often a kind of an optimisation condition, involving again concepts such as entropy, complexity, compactness, fuzziness, etc. [14, 40]. On the other hand, in the theory of fuzzy sets, T-transitive closures and T-transitive openings are well-known concepts that tend to minimize, in some precise mathematical sense, the deviation between the initial similarity matrix and a neigbouring T-equivalence relation. Herein, T denotes a triangular norm [26], whereas a special role is played by the minimum operator for the hierarchical representation of the fuzzy relations. Typically, single linkage cluster-

ing generates the min-transitive closure, whereas complete linkage clustering generates a min-transitive opening of the given similarity matrix.

Many algorithms exist for generating the T-transitive closure of an $(n \times n)$ similarity matrix, the most efficient algorithms being of time order $\mathcal{O}(n^2)$ if the triangular norm is the minimum operator, and of order $\mathcal{O}(n^3)$ otherwise. Much less efforts have been devoted to the problem of generating one or more T-transitive openings of a similarity matrix. In the present context, we restrict ourselves to the particular problem of generating one or more of the min-transitive openings of a similarity matrix. We will derive a new algorithm that turns out to be very efficient compared to other algorithms. In particular, we will illustrate that the min-transitive opening obtained by our method is in realistic applications, such as in the domain of microbiology, on the average closer to the original matrix than the min-transitive openings derived by other methods [5, 6]. Two more general algorithms will be proposed for the generation of a min-transitive approximation of a given similarity relation, which in general deviates less from the given similarity relation than its min-transitive closure and openings, and which is still guaranteed to be a similarity relation. Since these new algorithms are weight-driven, they can also be used to generate layer by layer the partition tree associated to the corresponding min-transitive approximation [7].

As a subdomain of microbiology, the ultimate goal of bacterial polyphasic taxonomy [39] is to classify the evolutionary diversity of the prokaryotes in a way that reflects as closely as possible the natural relationships between microorganisms. Practically, this classification is based on all available phenotypic, genotypic and phylogenetic characteristics of a sample of bacterial strains, supplying information on different taxonomic levels. Genotypic data is derived from the nucleic acids (DNA and RNA) present in the cells, whereas phenotypic data is derived from proteins and their functions, different chemotaxonomic markers and a wide range of other expressed features. Since researchers in this field deal with thousands of strains of bacteria, the time and space complexity of classification algorithms is indeed very important. Also, since similarity matrices are generally very large but their elements need not be stored with very high precision (e.g. round off to the second decimal yielding 101 possible values), many matrix elements will have the same value. We will bear this in mind when producing artificial data to test the quality of algorithms.

The outline of the present chapter is as follows. We set off with reviewing some characteristics of fuzzy equivalence relations in section 3.2, followed by an overview in section 3.3 of some of the methods found in the literature for constructing the T-transitive closure in a performant way. Section 3.4 goes deeper into the general properties of min-transitive openings and their associated partition trees, from which we will derive a new algorithm for generating a single min-transitive opening that is in general close to the original similarity matrix compared to its other min-transitive openings. Finally, section 3.5 reviews some of the work done on the generation of general T-transitive approximations, where we present two new weight-driven algorithms that produce approximative min-transitive similarity relations that deviate less from the given similarity relation than are its min-transitive closure and openings.

3.2 Equivalence relations

The notion of a crisp equivalence relation (i.e. a reflexive, symmetric and min-transitive binary relation) is a simple, yet important concept frequently encoutered in many mathematical theories. From a practical point of view, however, equivalence relations are of limited use only, as they do not allow to express degrees of relationship, for instance proximity or similarity degrees. This has led researchers already at an early stage to the theory of *fuzzy sets*, and in particular to the calculus of *fuzzy relations* [44]. Let us therefore first recall some basic notions concerning similarity relations, triangular norms and partition trees.

If $X = \{x_1, x_2, \ldots, x_n\}$ is a finite universe with dimension n, an *equational theory* (also called a *similarity model* or *proximity model* in some contexts) can be imposed on the universe X by means of a *binary fuzzy relation* R.

Definition 3.2.1 *A **binary fuzzy relation** R on a universe X is an $X^2 \to [0,1]$ mapping, where the value $R(x,y)$ is a quantitative expression of the degree of relationship between two members x and y of the universe X.*

As the issue of the current chapter is restricted to binary fuzzy relations, we will drop the adjective *binary* further on in the text. A fuzzy relation R on a finite universe X of cardinality n might be represented by means of the $n \times n$ matrix $A_R \equiv [a_{ij}]_R$, with matrix elements $a_{ij} \equiv R(x_i, x_j) \in [0,1]$. If R and S are two fuzzy relations on the same universe X, we say that R *is included in* S (S *contains* R), denoted $R \subseteq S$, if $(\forall (x,y) \in X^2)(R(x,y) \leq S(x,y))$. In matrix notation we also write $A_R \leq A_S$. If the fuzzy relation R is *reflexive*, i.e. for any $x \in X$ it holds that $R(x,x) = 1$, A_R has 1 everywhere on the diagonal. There also exists a weaker form of reflexivity, called *local reflexivity*.

Definition 3.2.2 *A fuzzy relation R on a universe X is **locally reflexive** if*

$$(\forall (x,y) \in X^2)(R(x,x) \geq R(x,y))\,. \tag{3.1}$$

In the matrix representation, this means that a diagonal element is not smaller than any element in the same row or column. A *similarity relation* R on a finite universe X is a fuzzy relation that is reflexive and symmetric.

Definition 3.2.3 *A **similarity relation** R on a (finite) universe X is a fuzzy relation on X which is*

(i) Reflexive: $(\forall x \in X)(R(x,x) = 1)$*;*

(ii) Symmetric: $(\forall (x,y) \in X^2)(R(x,y) = R(y,x))\,.$

The matrix representation A_R of a similarity relation R is called a *similarity matrix* (a symmetric matrix with elements in $[0,1]$ and 1's on the diagonal). In a graph-theoretical environment, a similarity relation R can be equivalently represented by means of a weighted undirected complete graph, with the set of elements of the finite universe constituting the vertices of the graph and the undirected edge connecting the vertices x and y carrying weight $R(x,y)$. This graph representation is more compact than the matrix representation, as the symmetry of the similarity relation dictates that edges of the graph can be made undirected. Moreover, given the reflexivity of similarity relations, loops can be removed from the graph representation, as they can all be implicitly regarded as 1.

Triangular norms (or *t-norms* for short) were introduced in the sixties by Schweizer and Sklar [37], as a means to generalize the triangle inequality to probablistic metric spaces. Since the eighties, they have been intensively applied for defining the intersection of fuzzy sets and to model the logical *and* in fuzzy logic. A t-norm is an increasing, commutative and associative binary operation on the unit interval $[0,1]$ which has 1 as neutral element [26].

Definition 3.2.4 *A mapping $T : [0,1]^2 \to [0,1]$ is called a **triangular norm** or **t-norm**, if the following conditions are met*

(i) Neutral element: $(\forall x \in [0,1])(T(x,1) = x)$;

(ii) Increasing: $(\forall (x,y,z) \in [0,1]^3)(x \leq y \Rightarrow T(x,z) \leq T(y,z))$;

(iii) Commutative: $(\forall (x,y) \in [0,1]^2)(T(x,y) = T(y,x))$;

(iv) Associative: $(\forall (x,y,z) \in [0,1]^3)(T(x,T(y,z)) \leq T(T(x,y),z))$.

In some contexts, $T(x,y)$ is also called the *T-product* of x and y, when T is representing a triangular norm. A triangular norm T is called *idempotent* if $(\forall x \in [0,1])(T(x,x) = x)$. The only triangular norm that is idempotent is the *minimum operator*

$$M(x,y) = \min(x,y)\,. \tag{3.2}$$

The minimum operator is also the largest triangular norm in the sense that $T(x,y) \leq M(x,y)$ for all $(x,y) \in [0,1]^2$ and any other triangular norm T. Other well-known examples of continuous t-norms are the *algebraic product* P,

$$P(x,y) = x\,y\,, \tag{3.3}$$

the *Łukasiewicz triangular norm* W,

$$W(x,y) = \max(x+y-1,0)\,, \tag{3.4}$$

and the *drastic product* Z, which is defined by

$$Z(x,y) = \begin{cases} \min(x,y) & \text{if } \max(x,y) = 1\,, \\ 0 & \text{otherwise}\,. \end{cases} \tag{3.5}$$

The drastic product is also the smallest amongst the family of triangular norms. Both the Łukasiewicz triangular norm and the drastic product have *zero divisors*, which means that

there exist $x, y > 0$ such that $T(x, y) = 0$. The four triangular norms that were introduced above, can be ranked in the following way

$$Z \leq W \leq P \leq M\,. \tag{3.6}$$

The inverse of a t-norm does not exist in general. A kind of inverse operator is nevertheless provided by its *residual implicator*.

Definition 3.2.5 *The **residual implicator** I_T of a t-norm T is a $[0,1]^2 \to [0,1]$ mapping defined by*

$$(\forall (x, y) \in [0, 1]^2)(I_T(x, y) = \sup\{z \in [0, 1] \,|\, T(x, z) \leq y\})\,. \tag{3.7}$$

In case of a continuous t-norm T and $y \leq x$, $I_T(x, y)$ is the greatest solution of the equation $T(x, z) = y$. Especially, we have that

$$I_M(x, y) = \begin{cases} 1 & \text{if } x \leq y\,, \\ y & \text{if } x > y\,, \end{cases} \tag{3.8}$$

$$I_P(x, y) = \begin{cases} 1 & \text{if } x \leq y\,, \\ \frac{y}{x} & \text{if } x > y\,, \end{cases} \tag{3.9}$$

$$I_W(x, y) = \min(1 - x + y, 1)\,, \tag{3.10}$$

and

$$I_Z(x, y) = \begin{cases} 1 & \text{if } x < 1\,, \\ y & \text{if } x = 1\,. \end{cases} \tag{3.11}$$

The residual implicator I_T is sometimes also called the *quasi-inverse* of the t-norm T [3]. For further reading on the properties of triangular norms, we refer to the work of Klement *et al.* [26].

T-transitivity is regarded as one of the most crucial properties that can be attributed to fuzzy relations, with T representing a triangular norm. The idea behind transitivity is that the degree of the interaction between two elements should not be less than the degree of any indirect chain containing other elements [18].

Definition 3.2.6 *For a given t-norm T, a fuzzy relation R on a universe X is called T-**transitive**, if for any $(x, y, z) \in X^3$ it holds that*

$$T(R(x, y), R(y, z)) \leq R(x, z)\,. \tag{3.12}$$

With the ranking of (3.6) in mind, we will also say for example that Łukasiewicz transitivity is a weaker form of transitivity than min-transitivity, as from the above definition it immediately follows that if a similarity relation is Łukasiewicz transitive it must equally be min-transitive. In analogy to the case of *crisp* relations (i.e. relations R where all elements $R(x, y)$ are in $\{0, 1\}$), a T-transitive similarity relation is also called a T-*equivalence*.

Definition 3.2.7 *A T-**equivalence** R on a finite universe X is a fuzzy relation on X, which is*

(i) Reflexive: $(\forall x \in X)(R(x,x) = 1)$ *;*

(ii) Symmetric: $(\forall (x,y) \in X^2)(R(x,y) = R(y,x))$ *;*

(iii) T-transitive: $(\forall (x,y,z) \in X^3)(T(R(x,y), R(y,z)) \leq R(x,z))$ *,*

where T represents a triangular norm.

If the property of symmetry is dropped from the definition of a T-equivalence, then the fuzzy relation is called a *T-preorder* (or sometimes a *T-quasi-order*). Note that some authors prefer to call fuzzy relations that conform to the properties of Definition 3.2.3 *proximity relations*, reserving the name *similarity relation* to describe fuzzy relations that satisfy the properties of Definition 3.2.7.

For the special case where the triangular norm is the minimum operator, it immediately follows from its definition that if R is a min-equivalence, then for any $(x,y,z) \in X^3$, two of the three elements $R(x,y), R(y,z), R(x,z)$ are equal while the third element is necessarily greater than or equal to the two other elements, or stated differently

$$(\forall (x,y,z) \in X^3)(\min\{R(x,y), R(y,z), R(x,z)\} = \text{median}\{R(x,y), R(y,z), R(x,z)\})\,. \tag{3.13}$$

Based on the definition of the *α-cut* of a fuzzy relation, an even more simplified representation of min-equivalences can be constructed.

Definition 3.2.8 *For a given $\alpha \in [0,1]$, the **cut** R_α **at cutting level** α of a fuzzy relation R on a universe X, is the crisp relation on X defined by*

$$(x,y) \in R_\alpha \;\Leftrightarrow\; R(x,y) \geq \alpha\,. \tag{3.14}$$

If R has matrix representation A_R, then the matrix representation A_{R_α} of R_α is given by

$$(a_{ij})_{R_\alpha} = \begin{cases} 1 & \text{if } a_{ij} \geq \alpha\,, \\ 0 & \text{if } a_{ij} < \alpha\,. \end{cases} \tag{3.15}$$

An important theorem states that a similarity relation R is min-transitive if and only if for every $\alpha \in [0,1]$ the cut R_α is min-transitive [44], hence an equivalence relation on X. The equivalence classes of R_α constitute a partition of X at cutting level α. With decreasing α, the equivalence classes tend to merge. The graph representation of this hierarchy of equivalence classes is called the *partition tree* of the min-equivalence. An alternative but equivalent representation consists of a node-weighted binary tree with its leaf nodes representing the individual objects. This latter representation is often called the *dendrogram* of a min-equivalence. Hereby, the degree of similarity between the objects x and y coincides with the weight of the least common ancestor in the binary tree of nodes x and y. Finally, for the dendrogram it holds that weights of the nodes on a path from a leaf node to the root are non-increasing, where the weights of the leaf nodes are regarded as 1. Hence, the root

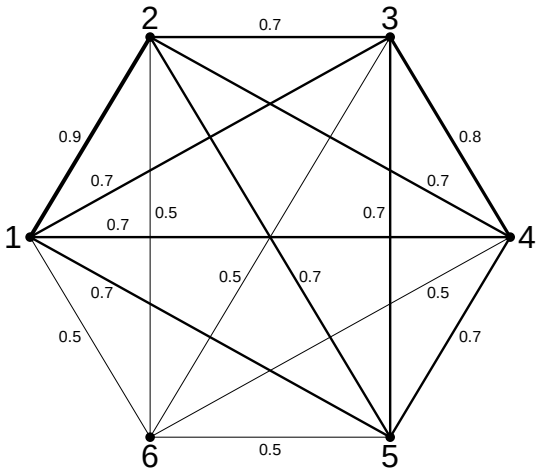


Figure 3.1: Weighted undirected complete graph representation of the min-equivalence R associated to the min-transitive similarity matrix A_R.

carries the lowest weight, which coincides with the smallest value of the min-equivalence. The tree representation of min-equivalences is thus more compact than their corresponding matrix or graph representations, due to the built-in hierarchical representation of the min-transitivity feature of the fuzzy relations.

As an illustration of the different representations of min-equivalences, let us consider the binary fuzzy relation R on a finite universe $X = \{1, 2, 3, 4, 5, 6\}$ with matrix representation

$$A_R = \begin{bmatrix} 1.0 & 0.9 & 0.7 & 0.7 & 0.7 & 0.5 \\ 0.9 & 1.0 & 0.7 & 0.7 & 0.7 & 0.5 \\ 0.7 & 0.7 & 1.0 & 0.8 & 0.7 & 0.5 \\ 0.7 & 0.7 & 0.8 & 1.0 & 0.7 & 0.5 \\ 0.7 & 0.7 & 0.7 & 0.7 & 1.0 & 0.5 \\ 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 1.0 \end{bmatrix}. \tag{3.16}$$

It can be easily shown that the fuzzy relation associated to the matrix A_R is reflexive, symmetric and min-transitive, hence a min-equivalence. Figure 3.1 shows the weighted undirected complete graph representation of the min-equivalence R associated to the min-transitive similarity matrix A_R. Any complete undirected subgraph composed from three nodes i, j, k and three edges with respective weights a_{ij}, a_{jk} and a_{ik}, is called the *(weighted) triangle* Δ_{ijk} of the graph. It can then indeed be easily checked that for every triangle within the graph representation of Figure 3.1, the two lowest edge weights are the same, as was stated in (3.13). In Figure 3.2 are shown the (unique) partition tree and an associated weighted binary tree of the min-equivalence R. Note, that in this case the node-weighted binary tree representation of R is not unique, since two internal nodes, one being parent of the other, carry the same weight. In fact, the other weighted binary tree representation is obtained by interchanging these two internal nodes. An equivalent (unique) representation with general trees and weights strictly decreasing on all paths from a leaf node to the root is also possible (see Figure 3.3). We will however restrict ourselves to binary tree representations in the rest of this chapter, given the general bifurcation procedure through

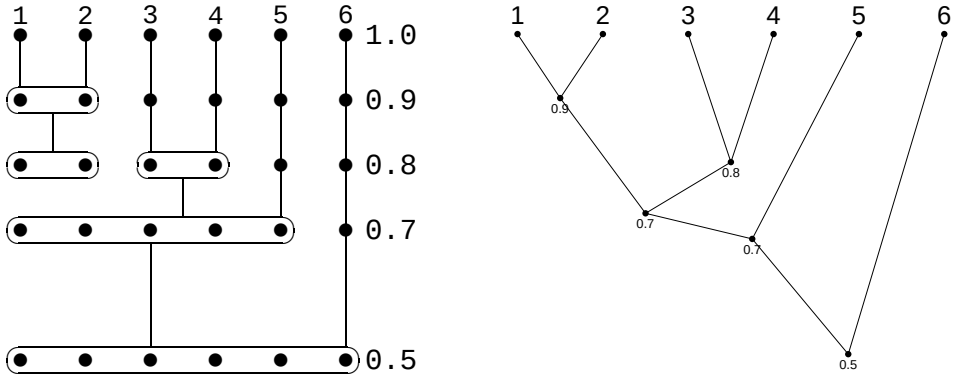


Figure 3.2: The partition tree (left) and a node-weighted binary tree (right) associated to the min-transitive similarity matrix A_R.

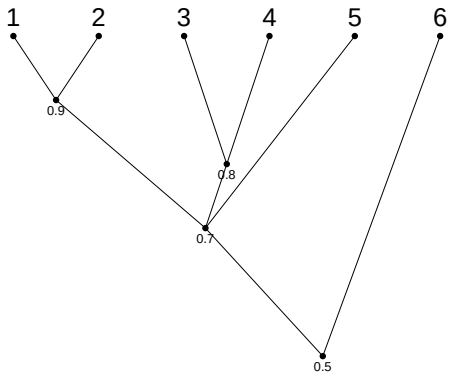


Figure 3.3: Unique node-weighted general tree associated to the min-transitive similarity matrix A_R.

which these trees are constructed in most cases. Due to the binary structure of a node-weighted tree associated to a min-transitive similarity matrix, each internal node possesses exactly two branches and two associated disjoint subsets of leaf nodes. These two subsets will be called the two *clusters*, say C_1 and C_2, associated to that internal node, whereas the internal node will be called the *least common ancestor* of the two clusters. In the example of Figure 3.2, for instance, the internal node closest to the root that carries weight 0.7 is the least common ancestor of the two clusters $C_1 = \{1, 2, 3, 4\}$ and $C_2 = \{5\}$ associated to this node. Also note the common practice to scale the position of the internal nodes of dendrograms according to their corresponding weights, which we have adopted for the representation of all trees in this chapter. This helps to visualize the homogeneity or heterogeneity of all clusters within the hierarchy, and simplifies the delineation of α-cuts for a given dendrogram. For the tree of Figure 3.3, one can for example easily read that the partition at cutting level $\alpha = 0.75$ is $\{\{1, 2\}, \{3, 4\}, \{5\}, \{6\}\}$.

A min-equivalence on a finite universe can thus be nicely visualized in a hierarchical manner by means of its partition tree. In particular, such a tree facilitates the interpretation of similarity relationships existing in a given set of objects and it is therefore not at all

surprising that partition trees, also called *hierarchical clustering trees*, have diverse applications. Essential for the existence of the partition tree representation of a min-equivalence, is its defining property of min-transitivity. In many circumstances, however, similarity relations do not naturally possess this property, and consequently min-transitivity then needs to be imposed artificially. This is for instance the case when the similarity relation is obtained through the use of a similarity measure (e.g. the Jaccard measure [24], the Dice measure [15], etc.) on the powerset of the set of objects under consideration, which generally results in a weaker form of T-transitivity or no transitivity at all [9]. Further on in this chapter we will look for ways of calculating min-transitive approximations that are generally close to their original similarity relation.

3.3 Transitive closure

From a practical point of view, the important question arises whether it is possible to find (if it exists) the smallest T-transitive fuzzy relation dominating a given fuzzy relation, or, in other words, whether it is possible to force T-transitivity by adding minimal values to the given fuzzy relation.

Definition 3.3.1 *A fuzzy relation $\hat{R}^T$ is called the T-**transitive closure** of a given fuzzy relation R, if and only if the following conditions are met*

(i) $\hat{R}^T$ *is T-transitive ;*

(ii) $R \subseteq \hat{R}^T$ *;*

(iii) If R' is T-transitive and $R \subseteq R'$, then $\hat{R}^T \subseteq R'$.

It is clear that the T-transitive closure, if it exists, must be unique. Indeed, if $\hat{R}_1^T$ and $\hat{R}_2^T$ both satisfy Definition 3.3.1 for a given fuzzy relation R, then (*iii*) implies $\hat{R}_1^T \subseteq \hat{R}_2^T$ and $\hat{R}_2^T \subseteq \hat{R}_1^T$. This supports the use of the definite article in the definition of the T-transitive closure. Moreover, it is shown that a fuzzy relation is T-transitive if and only if it coincides with its T-transitive closure [2]. A thorough investigation of the existence of the T-transitive closure of a fuzzy relation can be found in [2, 11], where it is proven that any fuzzy relation R on an arbitrary universe X has a T-transitive closure, for any t-norm T.

The T-transitive closure of a similarity relation R can thus be regarded as the smallest T-equivalence containing R. The studies on the existence of the T-transitive closure of fuzzy relations [2, 11] also came up with a procedure for the construction of the T-transitive closure for similarity relations through a series of $\sup$-T matrix multiplications, which performs (in the worst case) with $\mathcal{O}(n^3 \log n)$ time complexity and $\mathcal{O}(n^2)$ memory requirements. Recall that the $\sup$-T composition is defined in the following way.

Definition 3.3.2 *The $\sup$-T **composition** of two fuzzy relations R and S on the same uni-*

verse X is the fuzzy relation $R \circ_T S$ on X defined by

$$R \circ_T S(x, y) = \sup_{z \in X} T(R(x, z), S(z, y)) \,, \tag{3.17}$$

where T represents a triangular norm.

For finite universes, the supremum can be replaced by the maximum operator. If $A_R = [r_{ij}]$ and $A_S = [s_{ij}]$ are the respective matrix representations of two fuzzy relations R and S of cardinality n, then the matrix representation of the $\max$-T composition $R \circ_T S$ is given by

$$(R \circ_T S)_{ij} = \max_{1 \leq k \leq n} T(r_{ik}, s_{kj}) \,. \tag{3.18}$$

Using the definition of the $\sup$-T composition of fuzzy relations, the T-transitivity property can be formulated more concisely, in that a fuzzy relation is T-transitive if and only if

$$R \circ_T R \subseteq R \,. \tag{3.19}$$

We will use the notation $R^{(2)_T} := R \circ_T R$. As for finite universes X the $\max$-T composition is associative for any t-norm T, higher order T-powers for $k \geq 2$ can be defined unambiguously as

$$R^{(k)_T} := R^{(k-1)_T} \circ_T R = R \circ_T R^{(k-1)_T} \,. \tag{3.20}$$

It has been shown [2, 11] that the T-transitive closure $\hat{R}^T$ of a fuzzy relation on a universe X with cardinality n is given by

$$\hat{R}^T = \bigcup_{k=1}^{n} R^{(k)_T} \,, \tag{3.21}$$

where $\cup$ stands for the usual $\max$-based union of fuzzy sets. Also, the upper limit n can be lowered by one unit if R is locally reflexive. Given the fact that for locally reflexive fuzzy relations R on a universe X with cardinality n, it holds that

$$R \subseteq R^{(2)_T} \subseteq R^{(3)_T} \subseteq \ldots \,, \tag{3.22}$$

the T-transitive closure of a similarity matrix R can thus be calculated as $\hat{R}^T = R^{(n-1)_T}$. Practically, one computes $R_1^T = R^{(2)_T}, R_2^T = R^{(4)_T}, \ldots, R_k^T = R^{(2^k)_T}$ until $R_k^T = R_{k-1}^T$ or $2^k \geq n - 1$. The T-transitive closure is then given by $\hat{R}^T = R_k^T$. This procedure is known as the *matrix method*.

Many former studies have focused on the development of algorithms for the construction of the T-transitive closure, which perform better than the matrix method. Kandel & Yelowitz [25] introduced an $\mathcal{O}(n^3)$ column-row scanning algorithm that is basically a reformulation of the *Floyd-Warshall algorithm* [32, 43], originally designed to solve the all-to-all shortest path problem. By stating the Floyd-Warshall algorithm in abstract form, Feijs & van Ommering [17] established a more general framework of methods that includes the so-called *grid algorithm* [38]. The traditional treatment of the Floyd-Warshall algorithm for calculation of the T-transitive closure of an n-dimensional matrix $A = [a_{ij}]$

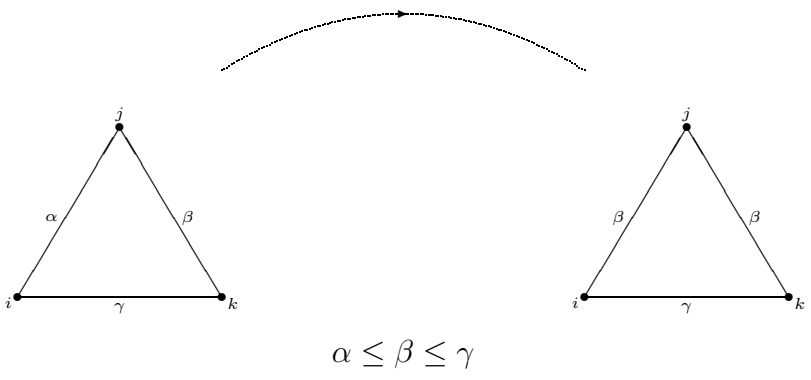


Figure 3.4: Local min-transitive closure operation.

is based on three nested for-loops

for i **from** 1 **to** n **do**
  for j **from** 1 **to** n **do**
    for k **from** 1 **to** n **do**
      $a_{jk} := \max(a_{jk}, T(a_{ji}, a_{ik}))$;
    endfor
  endfor
endfor

On output, the original matrix A is overwritten with the matrix representation of the T-transitive closure. In contrast to the matrix method, the Floyd-Warshall algorithm has the same time complexity and storage requirements as a single matrix multiplication, respectively $\mathcal{O}(n^3)$ and $\mathcal{O}(n^2)$. Naessens *et al.* [33] have introduced yet another algorithm for the calculation of the T-transitive closure of fuzzy relations with time complexity $\mathcal{O}(n^3)$ and $\mathcal{O}(n^2)$ memory needed.

For the specific case of the minimum operator as triangular norm, many algorithms have been described in the literature that break the barrier of $\mathcal{O}(n^3)$ time complexity. The local graph operation to transform a triangle that is not min-transitive into its min-transitive closure counterpart, is to raise the minimal weight to the level of the middle weight (Figure 3.4). Note that their is only one way to perform this local operation, which agrees to the unicity of the min-transitive closure. An algorithm with time complexity $\mathcal{O}(n^2)$ that is inspired by Primm's maximum weight spanning tree algorithm has been given by Dunn [16]. It has been recently reformulated by Kundu [27] who has also indicated how in the context of this algorithm the partition tree (called cluster-hierarchy tree) can be obtained within $\mathcal{O}(n^2)$ time. Lee [30] has equally succeeded to calculate the min-transitive closure in $\mathcal{O}(n^2)$ time, by making use of heaps. On the other hand, Larsen & Yager [28] have established an algorithm that makes direct use of an intermediate tree representation and has time complexity $\mathcal{O}(n^2 \log n)$ (the worst time complexity being $\mathcal{O}(n^3)$). In fact, this algorithm turns out to be equivalent with Kruskal's maximum weight spanning tree algorithm.

The so-called ascending-value method discussed in [19, 31] is yet another method with time complexity $\mathcal{O}(n^3)$. De Meyer *et al.* [13] have extended a previously derived weight-driven algorithm for computing the min-transitive closure and associated partition tree of a symmetric fuzzy relation in $\mathcal{O}(n^2)$ time. We conclude the issue of T-transitive closures with the remark that the min-transitive closure of a similarity relation on a finite universe is equivalent to *single linkage clustering* with the maximum as linkage function [36], for which an efficient matrix update scheme exists that needs exactly $n^2 - \frac{9}{2}n$ operations [1].

3.4 Transitive openings

3.4.1 T-transitive openings of a similarity relation

In analogy with the T-transitive closure, one can ask for some largest T-equivalence being included in a given similarity relation R. By definition, a *T-transitive opening* of a fuzzy relation R is a T-transitive fuzzy relation $\check{R}^T$ that is included is R, such that no other T-transitive fuzzy relation exists that contains $\check{R}^T$ and is itself included in R.

Definition 3.4.1 *A fuzzy relation $\check{R}^T$ is called a T-**transitive opening** of a given fuzzy relation R, if and only if the following conditions are met*

(i) *$\check{R}^T$ is T-transitive ;*

(ii) *$\check{R}^T \subseteq R$;*

(iii) *If R' is T-transitive and $\check{R}^T \subseteq R' \subseteq R$, then $R' = \check{R}^T$.*

A T-transitive opening of a fuzzy relation R is thus a T-transitive fuzzy relation that is maximally included in R. Therefore, T-transitive openings are also called *maximal T-transitive subrelations* in the literature [18]. Note that according to Definition 3.4.1, a similarity relation R can have more than one T-transitive opening, in contrast to the uniqueness of the T-transitive closure. If, for instance, R is a similarity relation of cardinality three with matrix elements $\alpha \leq \beta \leq \gamma$ and if the condition $\alpha \geq \min(\beta, \gamma)$ is violated, then the min-equivalences respectively found by lowering the middle weight β to α, or by lowering the largest weight γ to α, are the two min-transitive openings of R, the first being in general closer to R than the latter (more precisely, when $\beta < \gamma$). These local graph operations to transform weighted triangles that are not min-transitive into their min-transitive opening counterparts are shown in Figure 3.5, whereas the choice in local update transformations gives rise to a multitude of min-transitive openings that can generally been drawn for a given similarity relation that is not min-transitive.

In general, a T-transitive opening of a similarity relation is not necessarily symmetric, or in other words, not necessarily a T-equivalence. The only general purpose algorithm of

Fodor and Roubens [18], generating a T-transitive opening of a fuzzy relation, fails to preserve symmetry when applied upon a symmetric fuzzy relation, which makes the algorithm inapplicable for the construction of an approximative tree representation of a similarity relation. This algorithm recursively calculates a T-transitive opening of a given fuzzy relation R on the finite universe $X = \{x_1, x_2, \ldots, x_n\}$ of cardinality n in the following way

for j **from** 1 **to** n **do**
  $\check{R}^T(x_1, x_j) = R(x_1, x_j)$;
endfor

for i **from** 2 **to** n **do**
  for j **from** 1 **to** n **do**
    if $i > j$ **then**
      $\check{R}^T(x_i, x_j) = \min\{R(x_i, x_j), U(x_i, x_j), V(x_i, x_j)\}$;
    else
      $\check{R}^T(x_i, x_j) = \min\{R(x_i, x_j), V(x_i, x_j)\}$;
    endif
  endfor
endfor

with

$$U(x_i, x_j) = \min_{1 \leq k \leq n} I_T(\check{R}^T(x_j, x_k), R(x_i, x_k))\,,$$

and

$$V(x_i, x_j) = \min_{1 \leq k \leq i-1} I_T(\check{R}^T(x_k, x_i), \check{R}^T(x_k, x_j))\,.$$

Hence, this algorithm generates a T-transitive opening that has the same first row as its originating fuzzy relation. Through simultaneous transpositions of the rows and columns of the matrix representation associated to the fuzzy relation, the algorithm thus allows for the generation of multiple openings of the given fuzzy relation. Note also that in general the algorithm does not enable the calculation of all T-transitive openings for a given fuzzy relation, not even in the special case where T is the minimum operator, as there actually exist min-transitive openings that have no row in common with their original similarity relation.

The problem of generating all reflexive and symmetric min-transitive openings of a given similarity relation R is simple when R is injective, i.e. when all its non-diagonal matrix elements are mutually different. For this particular case, an algorithmic solution has been given by Leclerc [29], which progresses in the following manner. As all non-diagonal matrix elements are mutually different, there exists in the matrix representation $A_R = [a_{kl}]$ of the similarity relation a unique smallest matrix element a_{ij}, that expresses the similarity of the two most distant objects x_i and x_j within the finite universe X. The universe is accordingly partitioned into two subsets X_i and X_j, such that $x_i \in X_i$ and $x_j \in X_j$. This gives 2^{n-2} different possibilities to divide the remaining elements of $X \setminus \{x_i, x_j\}$ over the two subsets X_i and X_j. All matrix elements $a_{kl} = R(x_k, x_l)$ with $x_k \in X_i$ and $x_l \in X_j$ are then lowered to the value $a_{ij} = R(x_i, x_j)$. To finally generate a reflexive and symmetric min-transitive opening for the similarity relation R, the above procedure is recursively repeated on the restricted universes X_i and X_j, until singleton sets or sets with

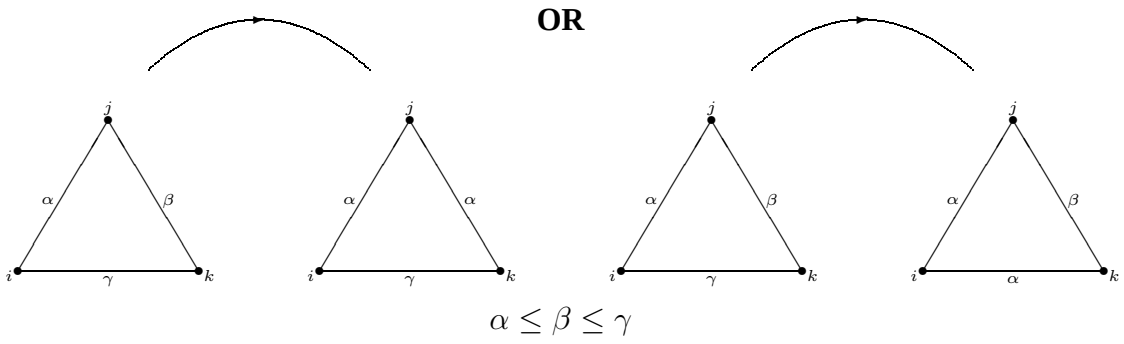


Figure 3.5: Local min-transitive opening operations.

only two elements are found. In theory, one could generate all min-transitive openings by means of the above procedure, and evaluate which openings are closest to the originating similarity relation, in terms of a predefined distance measure. However, Leclerc has also proven that there exist $(n-1)!$ different min-transitive openings for an injective similarity relation of cardinality n [29], which turns the generation of all different openings into an infeasible task, even for similarity relations of moderate cardinality. When operating upon a non-injective similarity relation R, the algorithm still generates min-equivalences that are included in R, but these are not necessarily min-transitive openings (however, all the min-transitive openings are generated). An exception occurs when R is crisp, in which case Leclerc's algorithm generates all min-transitive openings of R and only these [23]. In general, non-injective similarity relation thus have less than $(n-1)!$ different min-transitive openings.

In this section, we are primarily interested in the question whether algorithms exist that generate a single reflexive and symmetric min-transitive opening, which is relatively close to its originating similarity relation in comparison to other possible min-transitive openings of that similarity relation. As far as we know, following the divisive strategy of Leclerc, only heuristic algorithms exist for the generation of a min-transitive subrelation, which only result in a min-transitive opening in the majority of the cases. The classical algorithm to calculate a reflexive and symmetric min-transitive opening for a given similarity relation in an agglomerative way, is the *complete linkage clustering* algorithm [1, 36], which is also known as the *farthest neighbour clustering* algorithm. Further on, we will review this algorithm in more detail, before we present a new alternative algorithm for the calculation of a reflexive and symmetric min-transitive opening for a given similarity relation. But first we inspect some properties of the binary tree representation of reflexive and symmetric min-transitive openings.

3.4.2 The binary tree representation of min-transitive openings

According to the fact that every min-equivalence possesses at least one node-weighted binary tree representation, any algorithm for generating a reflexive and symmetric min-transitive opening of a similarity matrix $A = [a_{ij}]$, is essentially an algorithm that generates a node-weighted binary tree for which the following conditions must hold:

(W1) The node weights are non-increasing on the paths from a leaf node to the root.

(W2) The weight w of an internal node with associated clusters C_1 and C_2, is the minimum value of the matrix elements a_{ij}, where i and j run over all labels contained in C_1 and C_2 respectively, i.e.

$$w = \min_{i \in C_1, j \in C_2} a_{ij} \,. \tag{3.23}$$

(W3) For any three clusters C_1, C_2, C_3, such that the least common ancestor of $C_1 \cup C_2$ and C_3 is the parent node of the least common ancestor of C_1 and C_2, and such that both ancestral nodes carry the same weight, say λ, it must hold that

$$\min_{j \in C_2, k \in C_3} a_{jk} \leq \lambda \,. \tag{3.24}$$

Clearly, condition (W1) must hold for any node-weighted binary tree representation of a min-equivalence. Condition (W2) must hold for a binary tree representation of a symmetric min-transitive opening of a given similarity matrix A. Indeed, suppose that the internal node of the tree with associated clusters C_1 and C_2 carries the weight w' different from w in (3.23). If $w' > w$, then the tree cannot represent a min-transitive opening of A (since at least one element of the associated matrix is greater than the corresponding element of A), and if $w' < w$, then the tree obtained by replacing the weight w' by w represents a min-equivalence containing the associated matrix and being itself contained in A. Finally, let us suppose that condition (W3) were violated, in the sense that the binary tree representation of a min-equivalence A' contained in A possesses a subtree like the one depicted on the left-hand side of Figure 3.6, and where $\mu = \min_{j \in C_2, k \in C_3} a_{jk}$ with $\mu > \lambda$. Then, if this particular subtree is replaced by the subtree on the right-hand side of Figure 3.6, one obtains a node-weighted binary tree that is the representation of a min-equivalence that is contained in A and contains the matrix A', so that A' cannot be a reflexive and symmetric min-transitive opening of A.

We want to emphasize once more that the above conditions upon the binary tree representation of a reflexive and symmetric min-transitive opening of a similarity matrix are necessary, yet not sufficient conditions.

3.4.3 The complete linkage clustering algorithm

In general, two main strategies for constructing a valid node-weighted binary tree of a similarity relation can be distinguished, *i*) either by recursively splitting the set of leaf nodes

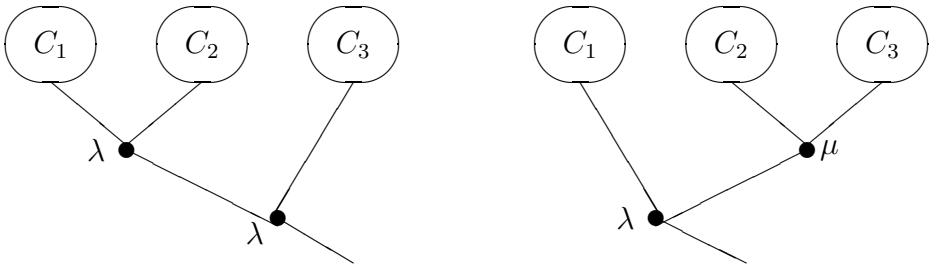


Figure 3.6: Tree conversion in case condition (W3) is violated.

and building the tree from the root onward (top-down approach, divisive clustering algorithms), or *ii*) by gradually linking nodes and building the tree from the leaf-nodes onward (bottom-up approach, agglomerative clustering algorithms). Leclerc's algorithm is based upon the first strategy, the complete linkage clustering algorithm upon the second. The new min-transitive opening algorithm that will be presented further on, also falls within the latter class of methods as it can be regarded as a modification of the complete linkage clustering algorithm. The genotypic and phenotypic bacterial features used for the construction of similarity matrices that are applied for inducing numerical taxonomies within the field of microbiology, are known to give more reliable measures for the close relationships between different microorganisms than for the more distant relationships (Figure 3.7, [39]). Therefore, intuitively, it seems obvious that the succes rate for generating close openings will be higher using an agglomerative strategy than with a divisive approach.

In order to situate more clearly the new algorithm with respect to complete linkage clustering, let us define for a given similarity matrix $A = [a_{ij}]$ on a finite universe X, the degree of similarity between two clusters C_1 and C_2 (in general, two non-empty disjoint subsets of X) as

$$s(C_1, C_2) = \min_{i \in C_1, j \in C_2} a_{ij} \,. \tag{3.25}$$

In the complete linkage clustering method, initially to each element i is assigned a cluster $C_i = \{i\}$ and in each new step two existing clusters are merged into a new cluster as follows: let $C_1, C_2, \ldots, C_m$ denote the actual clusters, then one determines the two clusters C_i and C_j for which the degree of similarity $s(C_i, C_j)$ is maximal. If there are several such maximal pairs, one pair is picked at random. The new cluster $C_i \cup C_j$ replaces the two clusters C_i, C_j, and all matrix elements a_{mn} and a_{nm} with $m \in C_i$ and $n \in C_j$ are lowered to the same value $s(C_i, C_j)$. Finally, this operational step is repeated until only one single cluster containing all the elements remains. Note that the single linkage clustering algorithm follows the same general agglomerative strategy as outlined above, with the only exception that the maximum operator is used in the definition of the cluster similarity in (3.25).

It is clear that at the end of this repetitive process, the obtained matrix is a reflexive and symmetric min-transitive opening of the given similarity matrix, as the conditions (W1–W3) are trivially satisfied for the node-weighted binary tree, which can be gradually constructed in parallel with the generation of new clusters. Moreover, the required ordering of

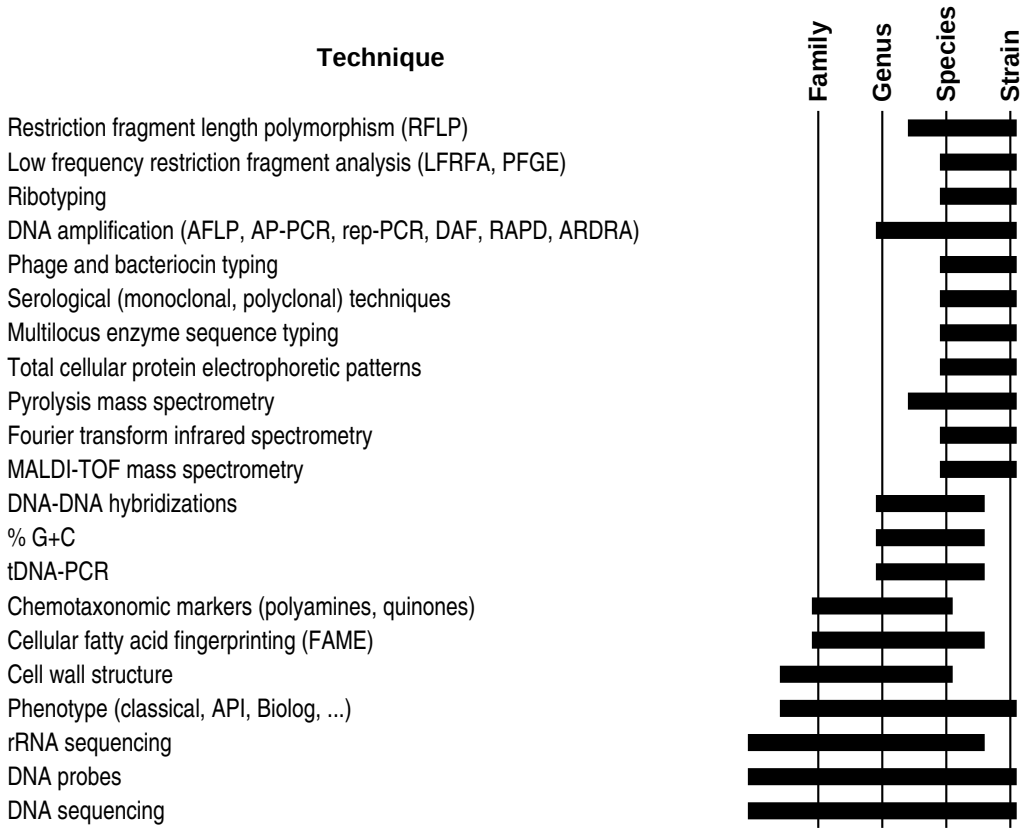


Figure 3.7: Schematic overview of the taxonomic resolution of some of the currently used techniques for comparing microorganisms (taken from [39] with courtesy of the author). PFGE, pulsed-field gel electrophoresis; ARDRA, amplified rDNA restriction analysis; RAPD, randomly amplified polymorphic DNA; AFLP, amplified fragment length polymorphism; MALDI-TOF-MS, matrix-assisted laser desorption ionisation time-of-flight mass spectrometry; PCR, polymerase chain reaction; FAME, fatty acid methyl esther.

weights on the paths from the leaf nodes to the root is realised since for any three clusters C_1, C_2 and C_3 it holds that

$$s(C_1 \cup C_2, C_3) = \min(s(C_1, C_3), s(C_2, C_3))\,. \tag{3.26}$$

By implementing the complete linkage clustering algorithm in an optimal way, it has $\mathcal{O}(n^2)$ space and time complexity [1], where n denotes the dimension of the given matrix (or, equivalently, the number of leaf nodes in the binary tree). Finally, note that for the selection of two clusters possessing the actual maximum similarity, at every step all pairs of clusters are taken into consideration. Intuitively, the associated binary tree is therefore expected to be usually rather well balanced.

3.4.4 A new min-transitive opening algorithm

In the approach of the new min-transitive opening algorithm, initially also to each element i a cluster $C_i = \{i\}$ is assigned, whereas in each step two clusters are merged into a single new one, until only a single cluster remains. But for the selection of the two merging clusters we do not longer consider all pairs of clusters, in the hope to speed up the performance of the algorithm. In fact, priority is given to the extension of one particular cluster by one element at each step. More precisely, the construction of the binary tree is initialised by selecting from the initial set of leaf nodes two different nodes, i and j, for which a_{ij} has the largest value in the matrix A, and to merge them into the cluster $C = \{i, j\}$. From then onwards, a typical intermediate situation is the one where just one cluster C contains more than one element, and we search for the element not contained in C that is the most similar to C, in other words the element $k \notin C$ for which $s(C, \{k\})$ reaches a maximum value λ. If that maximum is attained for only one k, then C is expanded into the new cluster $C \cup \{k\}$, which means that in the binary tree representation an internal node is created that is associated to the two clusters C and $\{k\}$ and carries the weight λ, whereas in the matrix representation all elements a_{ik} and a_{ki} with $i \in C$ are lowered to the same value λ. In the case that in each consecutive step the maximum is attained by a single element k, the final binary tree is a skew tree (sometimes called a spinal or caterpillar) with node weights strictly decreasing on the paths from a leaf node the root. Also, the associated matrix is a reflexive and symmetric min-transitive opening of A, due to property (3.26) (with C_2 and C_3 singletons).

If, however, there is a step in which the maximum $\lambda = s(C, \{k\})$ is attained for more than one $k \notin C$, say for the set $K = \{k_1, k_2, \ldots, k_m\}$, then it is not allowed (unlike in the complete linkage clustering algorithm) to pick one k_i at random to join C. Indeed, if in the next step one of the remaining k's, say k_j, is again candidate for the merging process and $s(C \cup \{k_i\}, \{k_j\}) = \lambda$, then condition (W3) would be violated if $a_{ij} > \lambda$ and the finally obtained binary tree would not be associated to a min-transitive opening of A. To solve this problem and to guarantee that condition (W3) will hold, we propose in this case to apply in a recursive way the tree constructing mechanism explained so far, its application now being restricted to the subset of nodes $K = \{k_1, k_2, \ldots, k_m\}$, and this as long as the weights carried by the newly created internal nodes are greater than or equal to λ. If such a tree exists (with root node carrying weight not smaller than λ), the cluster formed by the leaf nodes of this tree is in the next step merged at level λ with the cluster C, and the method continues as before; if no such tree exists, then it is allowed to pick at random one node from the set $\{k_1, k_2, \ldots, k_m\}$ and to merge this single node with the cluster C at cutting level λ.

A more detailed description of the new algorithm is presented in the recursive pseudocode procedure `GrowTree` shown in Figure 3.8, which concerns the construction of the node-weighted binary tree associated to a min-transitive opening. In fact, for an $(n \times n)$ similarity matrix A the node-weighted binary tree representation of a reflexive and symmetric min-transitive opening of A is generated by `GrowTree`$(A, S, L, 0)$, where $S = \{1, 2, \ldots, n\}$ denotes the initial (leaf) node set and the threshold value λ is initially set to 0. On exit of `GrowTree`, L is the subset of leaf nodes that are contained in the tree with

root weight not strictly smaller than λ. Note, however, that in the code only the creation of the tree nodes is explicitly mentioned whereas the creation of pointers to parent nodes, essential for the complete description of the tree structure, is left out. The main reason is that for building a min-transitive opening matrix $\check{A}$ of A, we don't need this complete tree description. Indeed, each time a new internal node is created in `GrowTree`, we simply have to attribute to all matrix elements $\check{a}_{ij}$ with i and j respectively in the leaf node sets of the two linked branches, as final value the weight of that internal node.

From a comparison of the new min-transitive opening algorithm to the complete linkage clustering algorithm, it is apparent that the average space and time-complexity of the former is not worse than that of the latter. It is even expected that the multiplicative constant can be made smaller for the new opening algorithm than for complete linkage clustering algorithm. This has been confirmed by the many experiments that we have carried out.

3.4.5 Numerical example

Let us illustrate the new min-transitive opening algorithm by means of the following example, where the input matrix A is given by

$$A = \begin{bmatrix} 1.0 & 0.9 & 0.6 & 0.4 & 0.3 & 0.4 \\ 0.9 & 1.0 & 0.7 & 0.3 & 0.4 & 0.3 \\ 0.6 & 0.7 & 1.0 & 0.8 & 0.4 & 0.4 \\ 0.4 & 0.3 & 0.8 & 1.0 & 0.7 & 0.2 \\ 0.3 & 0.4 & 0.4 & 0.7 & 1.0 & 0.5 \\ 0.4 & 0.3 & 0.4 & 0.2 & 0.5 & 1.0 \end{bmatrix}. \tag{3.27}$$

Initially, the procedure `GrowTree` is called with the complete set of leaf nodes $S = \{1, 2, 3, 4, 5, 6\}$ and the threshold value $\lambda = 0$ as parameter values. The maximal weight 0.9 for all pairs of different nodes taken from the set S is only reached for the nodes 1 and 2, so that these nodes are merged into the binary tree (a) shown in Figure 3.9. For the remaining nodes, the maximal weight $s = 0.6$ is attained for the singleton set $K = \{3\}$. As a result, node 3 can be safely merged with the existing tree (a), leading to dendrogram (b) of Figure 3.9. In the next step however, the maximal weight $s = 0.3$ is found for all members of the subset of leaf nodes $K = \{4, 5, 6\}$. Accordingly, the procedure `GrowTree` is recursively called with the subset of leaf nodes $S = \{4, 5, 6\}$ and the threshold value $\lambda = 0.3$ as parameters. In this recursive call on the restricted universe, the maximal weight 0.7 is reached for the nodes 4 and 5, so that the dendrogram (c) of Figure 3.9 is constructed. The last remaining node 6 cannot be merged with this dendrogram, as the resulting weight would be 0.2, which is lower than the threshold value $\lambda = 0.3$ of the recursive call. This means that the recursive call is exited, and the two dendrograms (b) and (c) are merged into the new tree (d) shown in Figure 3.9. Finally, node 6 is merged with the previously constructed tree (d) at level 0.2, althus resulting in the dendrogram (e) that represents the binary tree representation of an opening of the similarity matrix (3.27).

In Figure 3.10 the left and right node-weighted binary trees are the ones associated to the min-transitive openings generated respectively by the new opening algorithm and by

`GrowTree` (A, S, L, λ)

Input : A : similarity matrix
S : subset of leaf nodes
λ : threshold value

Output on exit : L : subset of leaf nodes that are contained in the tree with root weight not strictly smaller than λ

begin
$L := \emptyset$
if $\#S \leq 1$ **then** Exit **endif**
Select $(i, j) \in S^2$, $i \neq j$, such that $(\forall (l, m) \in S^2, l \neq m)\,(a_{ij} \geq a_{lm})$
if $a_{ij} < \lambda$ **then** Exit **endif**
Create parent of $\{i\}$ and $\{j\}$: weight a_{ij}
$L := \{i, j\}$
10:
Calculate $t_q = \min_{p \in L} a_{pq} \quad \forall q \in S \setminus L$
$s := \max_{q \in S \setminus L} t_q$
$K := \{k \mid t_k = s\} \equiv \{k_1, k_2, \ldots, k_n\}$
if $s < \lambda$ **then** Exit
else
if $K = \{k_1\}$ **then**
Create parent of L and $\{k_1\}$: weight s
$L := L \cup \{k_1\}$
if $S \neq L$ **then** Goto 10 **endif**
else
GrowTree (A, K, P, s)
if $P = \emptyset$ **then**
Create parent of L and $\{k_1\}$: weight s
$L := L \cup \{k_1\}$; Goto 10
else
Create parent of L and P: weight s
$L := L \cup P$;
if $S \neq L$ **then** Goto 10 **endif**
endif
endif
endif
end

Figure 3.8: Pseudo-code of the recursive procedure `GrowTree`, which is the basic building block for a new algorithm that calculates a reflexive and symmetric min-transitive opening and associated binary tree representation for a given similarity relation with similarity matrix A.

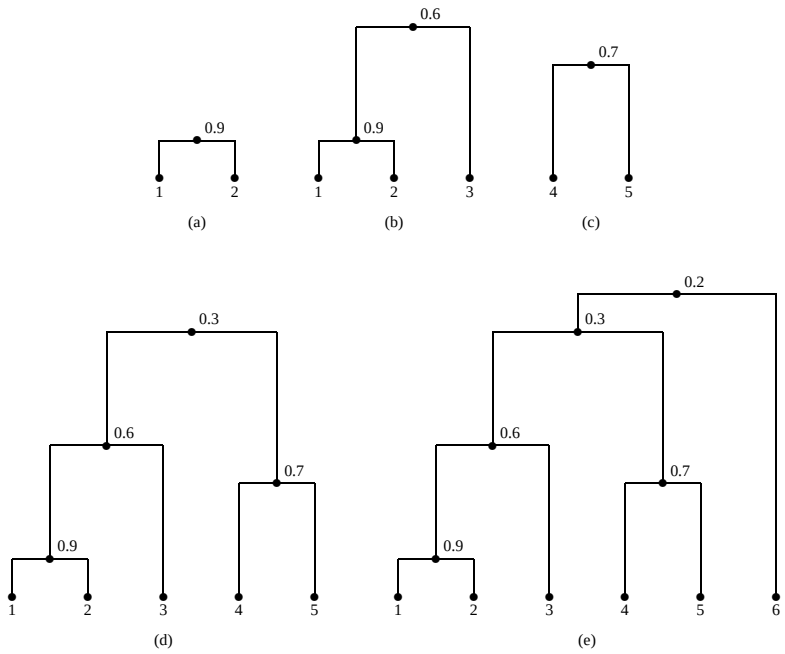


Figure 3.9: Intermediate trees generated by the `growTree` procedure during the construction of the min-transitive opening of the similarity matrix given in (3.27).

the complete linkage clustering algorithm. With the new opening algorithm this is the only tree that can be generated, whereas the complete linkage clustering algorithm yields in one of the intermediary steps a choice between two options (more precisely, also the tree with the branches at the nodes weighted 0.3 and 0.2 interchanged could have been generated). Although in this example the number of leaf nodes is small, one can easily observe that the tree generated by the new opening algorithm is indeed skewer than the complete linkage tree.

For the sake of completeness, we want to draw the attention upon the fact that the matrix A_R in (3.16) is nothing but the min-transitive closure of the matrix A used in this example. Note that the node-weighted binary tree associated to A_R and given in Figure 3.3, is structurally different from both binary trees in Figure 3.10.

It seems impossible to predict theoretically whether the new opening algorithm, which favours the construction of a skew binary tree, yields in general a min-transitive opening that is closer or further away from the original matrix than is the min-transitive opening obtained by the complete linkage clustering algorithm, which favours the construction of a more balanced binary tree. In order to see whether a definite pattern can be recognized, we need to carry out more experiments.

3.4.6 Measurement of average deviations

We can now ask the question which of the alternative opening algorithms has resulted in the min-equivalence that is closest to the original similarity relation, and how the different min-transitive openings score with respect to the min-transitive closure. After all, it is not

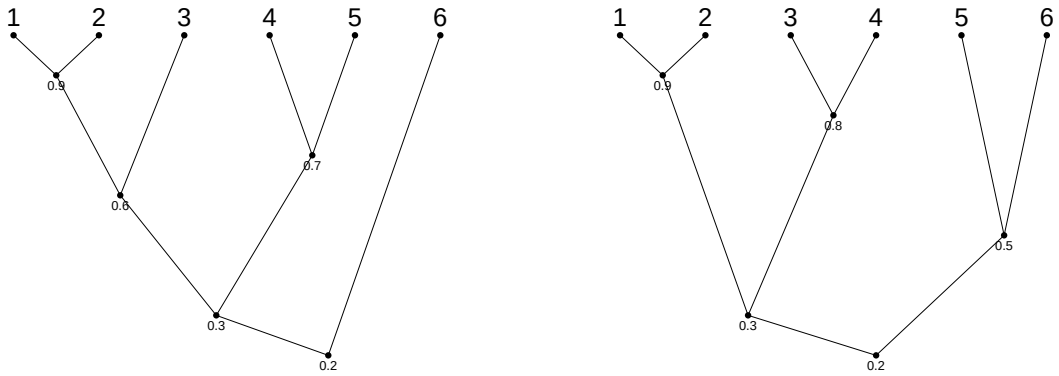


Figure 3.10: The dendrogram generated by algorithm `GrowTree` (left) and the dendrogram generated by the complete linkage clustering algorithm (right) for the same input matrix (3.27).

all that unnatural to search for the min-equivalence that has the smallest deviation from the genuine similarity model. Therefore, we first need to choose a measure for expressing the distance $d(A, B)$ between two n-dimensional similarity matrices $A = [a_{ij}]$ and $B = [b_{ij}]$. A first option is to calculate the normalized l_1-distance between the two similarity matrices, which is given by

$$l_1(A, B) = \frac{2}{n(n-1)} \sum_{1 \leq i < j \leq n} |a_{ij} - b_{ij}| \,. \tag{3.28}$$

Clearly $l_1(A, B)$ represents the average difference between two non-diagonal elements a_{ij} and b_{ij}, where the calculation of the average is restricted to the upper-diagonal elements of the matrices, given the symmetry and reflexivity properties of the similarity matrices we work with. An analogous normalized measure can be derived from the Euclidean distance or l_2-distance, in the following way

$$l_2(A, B) = \sqrt{\frac{2}{n(n-1)} \sum_{1 \leq i < j \leq n} (a_{ij} - b_{ij})^2} \,. \tag{3.29}$$

Note that both distance measures given above fall within the unit interval $[0, 1]$. For the comparison of min-transitive openings and closures, where no new matrix element values are introduced with respect to the original similarity matrix, and thus the precision of the matrices is not affected, we will opt for the l_1-distance given in (3.28) as the measure for determining the deviations during the calculation of approximative min-equivalences.

For the similarity matrix A in (3.27), we obtain that the reflexive and symmetric min-transitive opening generated by the new opening algorithm and possessing the left tree of Figure 3.10 as associated binary tree is at distance 0.113, the reflexive and symmetric min-transitive opening generated by the complete linkage clustering algorithm and possessing the right tree of Figure 3.10 as associated binary tree at distance 0.153, and the reflexive and symmetric min-transitive closure matrix A_R in (3.16) at distance 0.166 from the input matrix A. In this particular example, the new opening algorithm thus generates a min-transitive opening that is closer to the original matrix than both the min-transitive opening generated by means of the complete linkage clustering algorithm and the unique

min-transitive closure. The question naturally arises whether this is a general trend or just a coincidence.

In order to check this out, we have compared five algorithms for generating a reflexive and symmetric min-transitive opening of a given similarity matrix: two based on linkage, namely the complete linkage algorithm (CL) and the new opening algorithm (NA), and three based on splitting, namely three variants of Leclerc's algorithm, named L1, L2, L3 and corresponding to different splitting heuristics for generating one particular opening out of all openings.

More precisely, Leclerc's splitting heuristics are defined primarily for the case that all relational elements are mutually different (hence, all elements in the upper triangle of the given similarity matrix, or equivalently, all weights in the graph of the similarity relation, are different). For each of these variants, at each step a node set K is split into two disjoint subsets K_1 and K_2. The two nodes i and j in K for which a_{ij} actually reaches a minimum are separated, i being attributed to K_1 and j to K_2. The three algorithms (L1, L2, L3) differ in the way in which the other nodes belonging to K are divided over K_1 and K_2, respectively. With variant L1 of the algorithm, node $k \in K$ goes to K_1 if $a_{ik} > a_{jk}$, and to K_2 otherwise. In algorithm L2, the minimal spanning tree in the weighted graph associated to the node set K is constructed (note that (i, j) is an edge in that tree), and all neighbours in the tree of a node attributed to K_1 are attributed to K_2, and vice versa. Finally, in algorithm L3 a maximal spanning tree in the graph associated to K is constructed, the tree is split into two subtrees by discarding the edge of minimal weight and the nodes of the two subtrees are attributed to K_1 and K_2, respectively. One can easily extend each of these heuristics to the case where equal (non-diagonal) matrix elements occur, but it should be once more emphasized that then L1, L2 and L3 do not always yield a min-transitive opening.

A first basic experiment consisted in executing the 5 algorithms on a same arbitrary n-dimensional similarity matrix. By a random n-dimensional similarity matrix we mean that its independent elements (for instance the elements in the upper triangular part of the matrix) are uniformely selected from the set of numbers $\{j \cdot 10^{-N} \mid j = 0, 1, \ldots, 10^N\}$ with N a fixed integer. $1 + 10^N$ represents the maximum number of different values in the matrix, hence N will be called the *precision* of the matrix hereafter. Matrices generated with the above procedure are called matrices of type-1 in the context of this chapter, and denoted by the subscript t_1. For a fixed dimension n and precision N, the 5 algorithms are applied to the random similarity matrix A_{t_1}, and each time the distance between A_{t_1} and the generated min-transitive opening $\check{A}_{t_1}$ is calculated. The experiment is then repeated 100 times and the mean distances are computed for every algorithm. In Figure 3.11 these mean distances are plotted for $n = 10$ as functions of the precision of the input matrix. Figure 3.12 shows the results for the case $n = 100$.

From these figures we first notice that the average difference between non-diagonal elements is remarkably high and even approaches 0.5 for high-dimensional random similarity matrices with many different elements. This is apparently due to the fact that random matrices violate the transitivity condition so badly that the original elements need to be drastically lowered in order to obtain a min-transitive opening. The splitting algorithm L2

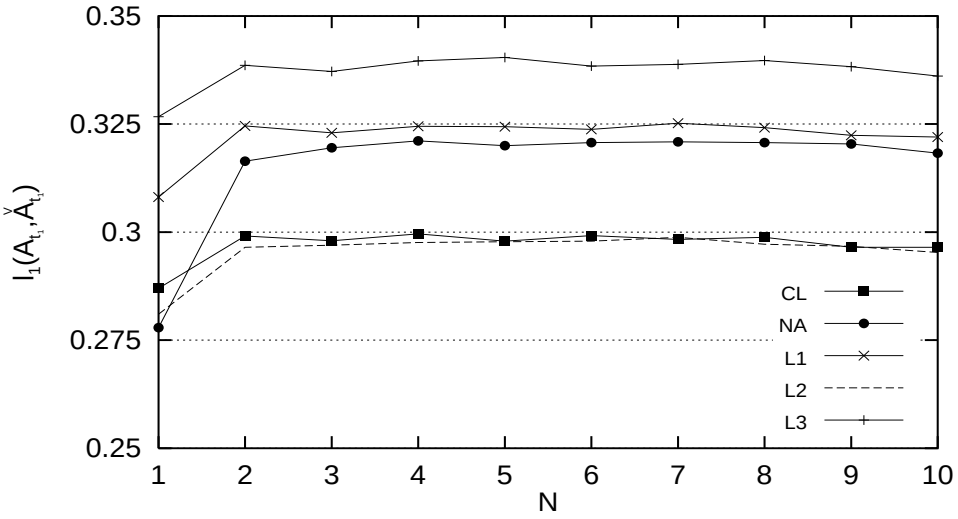


Figure 3.11: Comparison of five min-transitive opening algorithms acting upon 10-dimensional random similarity matrices (type-1).

is clearly the method that, compared to the other algorithms, generates the opening that is the least deviating from the original similarity matrix, although for smaller dimensions the difference with the complete linkage clustering algorithm CL, which in contrast to L2 always generates a min-transitive opening, becomes irrelevant. Finally, for this type of random matrices, the new opening algorithm NA cannot compete with the complete linkage algorithm, except for the case where many matrix elements have the same value (i.e. for very low precision N).

Since random matrices are not the kind of matrices that stand model for the similarity matrices encountered in practical applications, we have set up a second series of experiments in the following way. First we have generated an n-dimensional random vector $x = (x_1, x_2, \ldots, x_n)$ with elements uniformly selected from the set of numbers $\{j \cdot 10^{-N} \mid j = 0, 1, \ldots, 10^N\}$ with N a fixed integer. From this random vector we constructed the similarity matrix A with matrix elements $a_{ij} = a_{ji} = 1 - |x_i - x_j|$ for all $i < j$, and $a_{ii} = 1$. Matrices generated with the above procedure are called matrices of type-2 and denoted by the subscript t_2. This way of constructing an arbitrary similarity matrix A_{t_2} allows to attribute to N again the meaning of the precision of the matrix. It is also known that such a matrix A_{t_2} possesses a weak form of transitivity, belonging to the family of T-transitivity properties with T a triangular norm. More precisely, the matrix A is Łukasiewicz-transitive, which means that for all i, j, k it holds that

$$\max(a_{ik} + a_{kj} - 1, 0) \leq a_{ij} \,. \tag{3.30}$$

In Figure 3.13 are compared the average distances between the min-transitive openings generated by the five algorithms and an initial 10-dimensional similarity matrix of precision N, constructed as explained above. The average distance for fixed N has been computed by repeating the experiment 100 times. Figure 3.14 describes the average distances found for 100-dimensional similarity matrices. In both figures it is however difficult to distinguish the results for algorithm L2, as they almost completely coincide with those

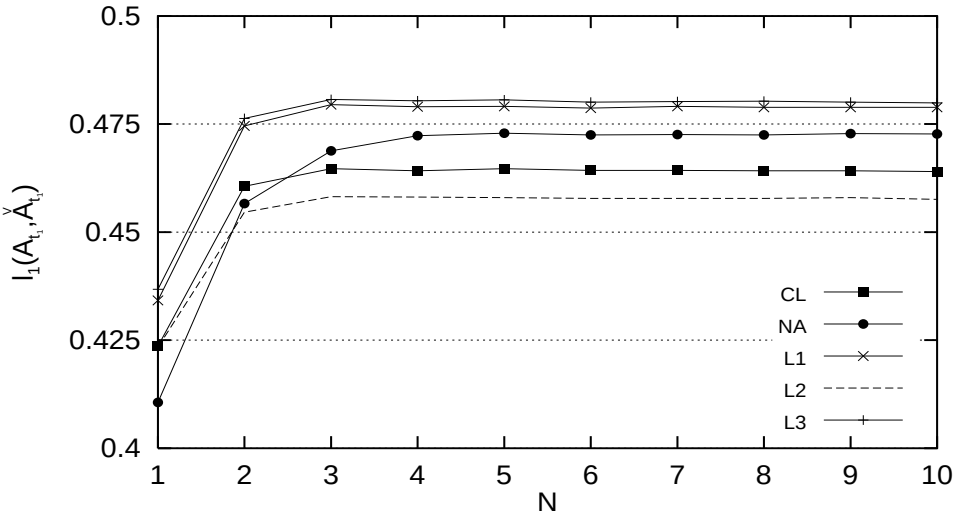


Figure 3.12: Comparison of five min-transitive opening algorithms acting upon 100-dimensional random similarity matrices (type-1).

obtained by algorithm L1. A first observation is that algorithm L3, which also before was outperformed by the other algorithms, now is, as far as distance to the initial matrix is concerned, performing very badly, whereas the difference between the other methods is quasi irrelevant. Just looking at average distance, the complete linkage algorithm might be considered the most preferable one, the only exception being the case of very low matrix precision, where the role of closest min-transitive opening algorithm is taken over by the new opening algorithm. However, taking also into consideration that the new algorithm runs faster than the complete linkage algorithm, a general preference for our algorithm can be justified.

A yet more decisive conclusion in the same direction can be drawn from a third series of experiments in which we try to simulate as close as possible realistic situations as the ones encountered in bacterial taxonomy. We now start by generating ten random vectors x_i $(i = 1, 2, \ldots, 10)$ with 120 components each, uniformly selected from the unit interval $[0, 1]$. These vectors are interpreted as fuzzy feature vectors and a 10-dimensional similarity matrix $S = [s_{ij}]$ is built by using the Jaccard similarity measure to express the similarity between the vectors, i.e.

$$s_{ij} = \frac{\sum_{k=1}^{120} \min(x_i(k), x_j(k))}{\sum_{k=1}^{120} \max(x_i(k), x_j(k))} .$$

One of the properties of the Jaccard measure is that the obtained similarity matrix S is always Łukasiewicz-transitive. In order to introduce again the concept of (variable) matrix precision, the (arbitrary) similarity matrix for which we compute a min-transitive opening by the different methods is not S, but instead the matrix A obtained by rounding off the elements of S conform the required precision N. If, for example, $N = 1$, then all elements are rounded off to the first decimal digit. Note that in this operation the symmetry of the matrix will not be destroyed, but obviously, the Łukasiewicz-transitivity might get lost. Matrices generated according to the above procedure are called matrices of type-3 and subscripted with the marker t_3.

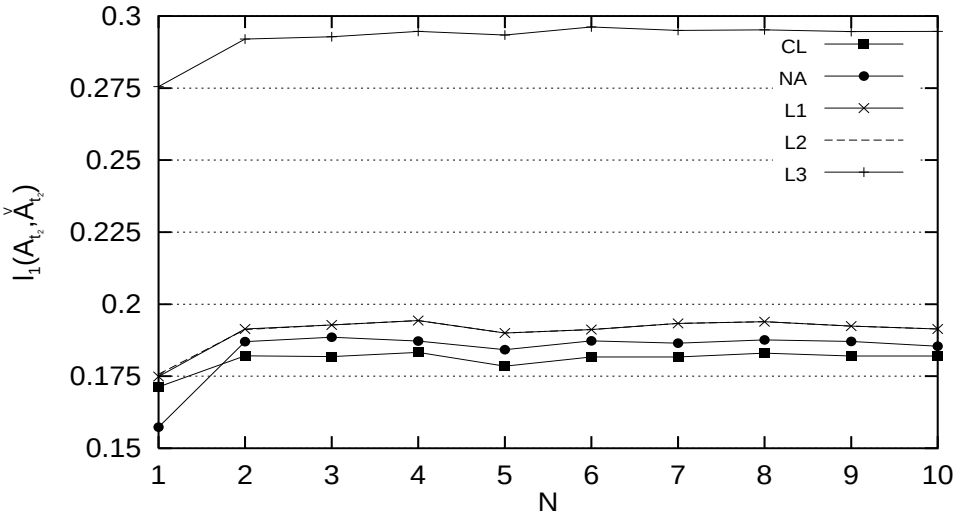


Figure 3.13: Comparison of five min-transitive opening algorithms acting upon 10-dimensional similarity matrices constructed from a 10-dimensional random vector (type-2).

In Figure 3.15 we compared the average distances between the min-transitive openings generated by the five algorithms for the arbitrary 10-dimensional Jaccard-based similarity matrices, where for each N averages have been computed by repeating the experiment 100 times. Note the peculiar behaviour of algorithm CL for precision $N = 1$. Figure 3.16 describes the average distances found for 100-dimensional arbitrary Jaccard-based similarity matrices, which are generated in the same way explained as before, with the only difference that we start from 100 (120-dimensional) random vectors.

From Figures 3.15-3.16, one observes that our method systematically accounts for the smallest mean distances, irrespective of the dimension n and of the precision N. Almost similar results are obtained when one starts from the Łukasiewicz min-transitive closure of a random similarity matrix (generated with the required precision).

3.5 Alternative transitive approximations

3.5.1 T-transitive approximations of a similarity relation

Approximative min-equivalences generated as min-transitive closures or min-transitive openings are restricted both in the sense that they do not introduce new values with respect to the original similarity matrix (as minimum and maximum are the only operations performed on the matrix elements) and they only strictly raise or lower the values of the original similarity relations in the respective cases. In this section we will therefore investigate what are the possibilities to generate T-transitive approximations in the case where it is allowed to simultaneously increase and decrease some of the matrix elements with respect to their initial values. Intuitively, it can be expected that this might generate T-equivalences that are generally closer to the original similarity matrix than are its T-transitive closure and

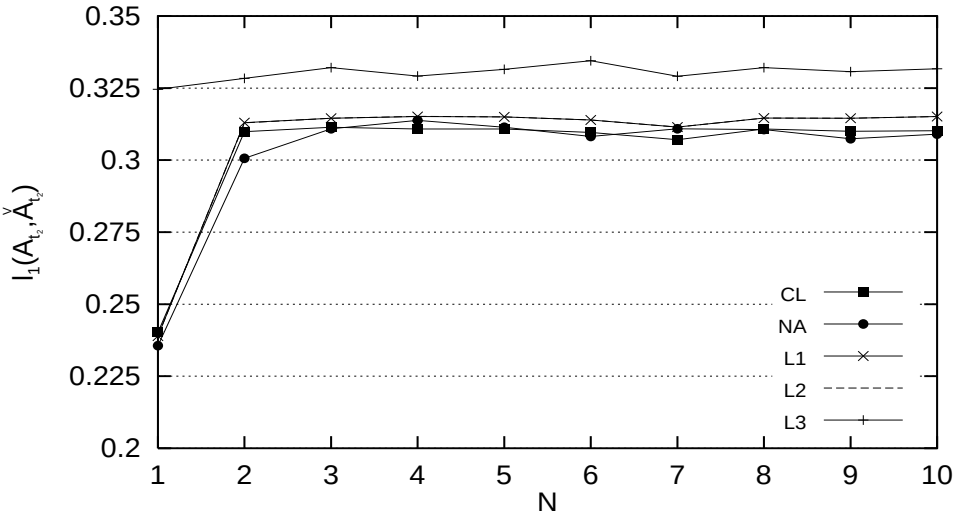


Figure 3.14: Comparison of five min-transitive opening algorithms acting upon 100-dimensional similarity matrices constructed from a 100-dimensional random vector (type-2).

T-transitive openings.

De Baets & De Meyer [10] recently developed the so-called *cascade method* for approximating a similarity relation on a finite universe by a min-equivalence, which tends to minimize the deviation between the initial similarity relation and the min-equivalence associated to the resulting partition tree. The key concept of this method is an alternation strategy of T-transitive closure and T-transitive opening operations, accomplished by using a parameterized family of triangular norms T_s, such as the Schweizer-Sklar family [37] or the Frank family [26], for which the t-norm T_s gradually progresses from the Łukasiewicz t-norm W to the minimum operator M. Remark that if a given t-norm T belongs to a known family of triangular norms that follows the conditions described above, the cascade method also enables the calculation of a T-transitive approximation for any given similarity relation, by stopping the alternation with a T-transitive closure, instead of progressing completely up to a min-transitive approximation.

With *UPGMA clustering* (unweighted pair-group method using arithmetic averages) one also usually obtains a min-equivalence of which some elements have been raised and other ones lowered with respect to their initial values. The obtained min-transitive approximation is therefore on the average closer to the initial relation than are, for instance, the approximations generated by the single linkage (closure) and complete linkage (opening) clustering algorithms. The UPGMA algorithm follows the same general agglomerative clustering strategy as the single linkage and complete linkage algorithms, as outlined in subsection 3.4.3. In the single linkage method each cluster is characterized by the shortest link needed to connect any member of the cluster to some other member of the cluster, whereas in the complete linkage method each cluster is characterized by the longest link needed to connect every member of a cluster to every other member. Instead of relying on extreme values as in these two cases, the UPGMA method evaluates the potential merging of two clusters C_1

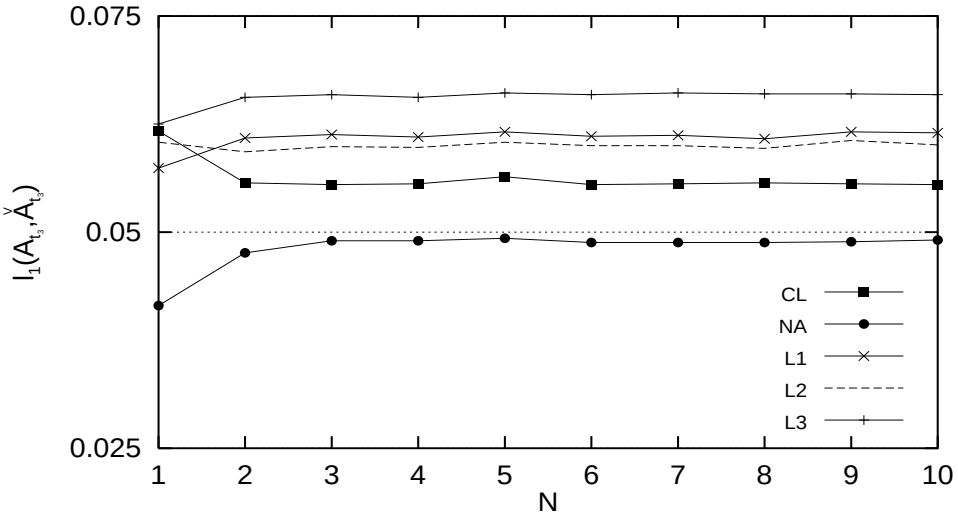


Figure 3.15: Comparison of five min-transitive opening algorithms acting upon 10-dimensional Jaccard-based similarity matrices (type-3) derived from 10 random vectors (with 120 components).

and C_2 in terms of the average degree of similarity between the elements of the two clusters

$$s(C_1, C_2) = \frac{\sum_{i \in C_1, j \in C_2} a_{ij}}{|C_1||C_2|}\,, \tag{3.31}$$

where $|C|$ represents the cardinality of the subset C. An alternative method based on average linkage consists of characterizing a cluster by the average of all links within it, so that the degree of similarity between two clusters C_1 and C_2 is regarded as

$$s(C_1, C_2) = \frac{2\sum_{i<j \in (C_1 \cup C_2)} a_{ij}}{|C_1 \cup C_2|(|C_1 \cup C_2| - 1)}\,. \tag{3.32}$$

The method resulting from the latter cluster similarity model is called *average linkage within the new group* in [1], whereas the UPGMA method is termed as *average linkage between merged groups*. As might be expected, both methods give results which are not radically different. There exist performant matrix update schemes for the implementation of both min-transitive approximation algorithms based on the average linkage cluster similarity models (3.31) and (3.32) [1]. Some textbooks [1, 36] mention hierarchical clustering techniques that are vulnerable for reversals, such as centroid linkage techniques and median linkage techniques. A reversal occurs when an object (or cluster) joins a cluster after the cluster has formed, but joins at a higher similarity level than that at which the cluster formed. In these cases the triangle inequality is not met, and with only few reversals the dendrogram starts to look like a wiring diagram for a color television set (Figure 3.17). Therefore, the simplicity of the hierarchical representation (and the min-transitivity) is lost, so that these kinds of hierarchical methods are not taken into consideration here.

An inherent weakness of UPGMA clustering and its variant method, however, is that the generated min-transitive approximation and the structure of its associated partition tree are not invariant under permutation of the objects. In order to get rid of this subjectivity

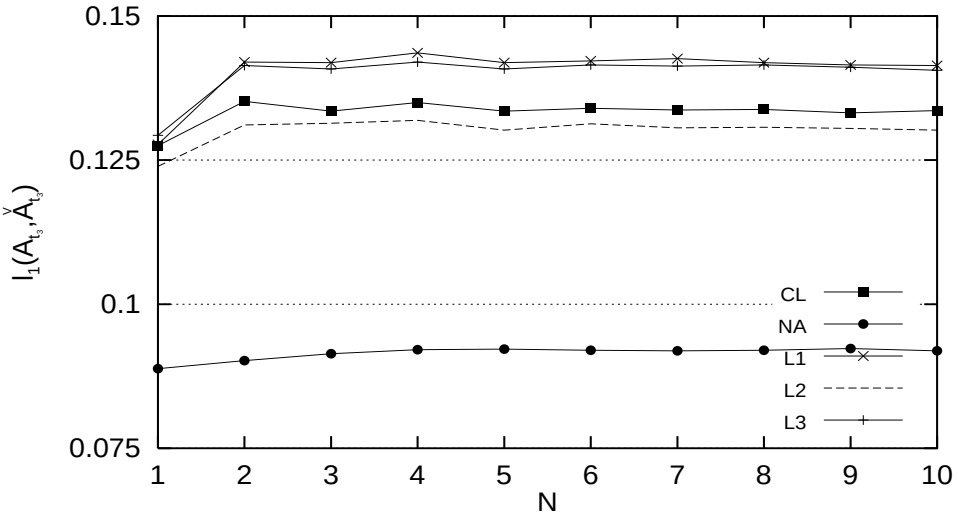


Figure 3.16: Comparison of five min-transitive opening algorithms acting upon 100-dimensional Jaccard-based similarity matrices (type-3) derived from 100 random vectors (with 120 components).

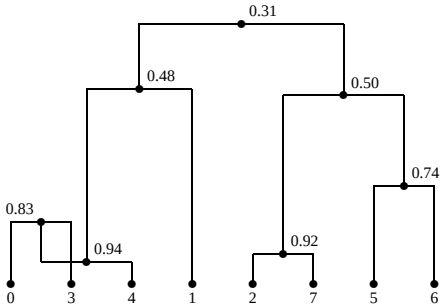


Figure 3.17: Example dendrogram with one reversal.

involved during the average linkage approximation procedures, we will propose in this section two new weight-driven algorithms for obtaining a min-transitive approximation of a similarity relation, and we report on the tests we have carried out to compare these algorithms to other existing approximation algorithms.

3.5.2 A first new min-transitive approximation algorithm

The UPGMA method cannot be modified in the same way as the complete linkage clustering algorithm was adapted to obtain the new min-transitive opening algorithm proposed in subsection 3.4.4. However, the procedure that was previously applied for the derivation of weight-driven algorithms for the computation of the T-transitive closure of an arbitrary fuzzy relation R on a finite universe [13, 33], has inspired us to modify the UPGMA method in a similar way. A detailed description of the first new weight-driven approximation algorithm that is established in this manner, is presented in the pseudo-code procedure APX1A of Figure 3.18.

APX1A (R, n)

Input : R, similarity relation
n, caridinality of R

Output : Min-transitive approximation $\tilde{R}$ of R, with $\tilde{R}(i,j) := w_{ij}$

begin
$S := \{1, 2, \ldots, n\}$
$V := \{w_{ij} = R(i,j) \,|\, (i,j) \in S^2\}$
$W := \{(i,j) \,|\, (i,j) \in S^2 \wedge i < j\}$
repeat
Select $(i,j) \in V$ such that
$(\forall\,(l,m) \in V)\ (w_{ij} \geq w_{lm})$
$\lambda := w_{ij}$
Build the maximal tree with node set P such that
$\{i,j\} \subseteq P$ and for any two adjacent nodes
$l, m \in P$ it holds that $w_{lm} = \lambda$
for all $(l,m) \in P^2 \wedge (l < m)$ **do**
$V := V \setminus \{(l,m)\}$
if $w_{lm} < \lambda$ **then** $w_{lm} = w_{ml} := \lambda$ **endif**
endfor
for all $k \in S \setminus P$ **do**
$t := \max_{l \in P} w_{kl}$
$s := \frac{1}{\#P} \sum_{l \in P} w_{kl}$
if $t \leq \lambda$ **then**
for all $l \in P$ **do** $w_{kl} = w_{lk} := s$ **endfor**
endif
endfor
until $V = \emptyset$
end

Figure 3.18: Pseudo-code of the procedure APX1A, which calculates a reflexive and symmetric min-transitive approximation and associated binary tree representation for a given similarity relation R of cardinality n.

In algorithm APX1A, every edge of the complete undirected complete graph representation of the given similarity relation is selected once, whereas the selection proceeds in descending order of the edge weights (note that weights change during the computation), the reason for which the algorithm is called a *weight-driven* algorithm. We will refer to the currently selected edge as the *pivot edge* further on in the text. The weight of the pivot edge, called the *pivot weight*, is denoted as λ and different consecutive pivot edges can have the same pivot weight. Once a pivot edge with weight λ is selected, two main operations are carried out. First we construct the largest tree containing the pivot edge and of which all the edges possess weight λ. It is called the *maximal tree* and its node set is denoted by P. The weights of the edges with end points in P that are smaller than λ are raised to the value λ. Remark that this local weight-lifting step is similar in nature to the min-transitive closure. Here, the procedure actually differs from the weight-driven algorithms for calculating the min-transitive closure, in that we cannot longer work with a single pivot edge in all cases, but rather with a cluster of pivot edges being all edges with end points in the maximal tree P. Secondly, for each node k not contained in P we consider all the triangular subgraphs consisting of that node k and two nodes in P. If none of the edges connecting k to a node in P is larger than λ, then all these edges obtain as same new weight the arithmetic mean of their initial values. In this way, all these triangular subgraphs have been made min-transitive, and the local weight modifications to achieve this min-transitivity are, according to the principle of least squares, the minimal possible ones. Let us call this part the *local optimization step* of the algorithm.

The proof that algorithm APX1A yields upon its termination a final weighted graph that is globally min-transitive, is based on the fact that the stepwise execution of the algorithm is equivalent to the layer-by-layer construction of the unique partition tree associated to this approximation. However, this equivalence can also be exploited to establish a highly efficient implementation of algorithm APX1A of which the time complexity is of optimal order $\mathcal{O}(n^2)$. Hence, the pseudo-code formulation of the algorithm shown in Figure 3.18 should be regarded as instructional only as it seriously differs from the optimal implementation.

Another important property of the weight-driven algorithm is that its interruption at the stage where all pivot edges with a same pivot weight, say α_s, have been considered, yields an intermediate similarity relation whose α-cut coincides for all $\alpha \geq \alpha_s$ with the α-cut of the min-transitive approximation. Hence, if not the entire approximation but only an α-cut of it is required, the algorithm can be interrupted once all edges carrying weight greater than or equal to α_s have been considered.

Since the arithmetic mean is used as a recipe for modifying weights, the algorithm APX1A somewhat resembles the UPGMA clustering algorithm. There are, however, two main differences to be emphasized. Firstly, when applied to a reflexive and symmetric crisp relation, algorithm APX1A generates a min-transitive crisp relation that collapses with the min-transitive closure due to the treatment of the maximal tree cluster. In fact, the local optimization step should be never executed in this case because it no longer produces any effect, taking only the average of zero-weighted edges. Hence, it generates for any given crisp similarity relation a unique partition (or classification). On the contrary, the application of the UPGMA algorithm upon a crisp relation yields, in general, a min-equivalence,

hence a full partition tree or dendrogram. In some contexts, this fuzzification of the equivalence relation might be unwanted. Secondly, the new algorithm APX1A generates an approximation that is less biased than the UPGMA approximation and at the same time is independent of the order in which ties are resolved in the selection of the clusters to be aggregated. Let us illustrate this difference on the trivial example where the graph of the given relation is a triangle with edge weights 1, 1 and 0. This is clearly the weighted graph of a relation that is not min-transitive. After UPGMA clustering, the zero weight is raised to 1/2 and one of the unit weights is lowered to 1/2, but which one of them depends on the order in which the objects are enumerated. On the other hand, algorithm APX1A raises the zero weight to 1 whatever the numbering of the objects is, thereby avoiding a bias for any one of the objects that are initially indistinguishable. This procedure is graphically represented in Figure 3.20.

3.5.3 Numerical example

In order to get a full understanding of the new min-transitive approximation algorithm, we will illustrate its procedure in a stepwise fashion on the similarity matrix A_R, given by

$$A_R = \begin{bmatrix} 1.0 & 0.3 & 0.6 & 0.8 & 0.1 & 0.6 & 0.2 & 0.2 \\ 0.3 & 1.0 & 0.6 & 0.2 & 0.7 & 0.6 & 0.8 & 0.8 \\ 0.6 & 0.6 & 1.0 & 0.5 & 0.4 & 0.9 & 0.5 & 0.5 \\ 0.8 & 0.2 & 0.5 & 1.0 & 0.0 & 0.5 & 0.1 & 0.1 \\ 0.1 & 0.7 & 0.4 & 0.0 & 1.0 & 0.4 & 0.8 & 0.8 \\ 0.6 & 0.6 & 0.9 & 0.5 & 0.4 & 1.0 & 0.5 & 0.5 \\ 0.2 & 0.8 & 0.5 & 0.1 & 0.8 & 0.5 & 1.0 & 0.9 \\ 0.2 & 0.8 & 0.5 & 0.1 & 0.8 & 0.5 & 0.9 & 1.0 \end{bmatrix} . \tag{3.33}$$

It can be easily checked that this similarity matrix is Łukasiewicz transitive, but not min-transitive, as for example the triangle Δ_{234} does not satisfy the condition given in (3.13). Initially, the set of candidate pivot edges reaching the maximal weight λ = 0.9 is equal to $\{a_{36}$,$a_{78}\}$. If the lexicographically first edge a_{36} is chosen as starting point, a maximal tree is found with P = {3,6} as the only nodes. For all nodes outside the set P, the local optimization step of algorithm APX1A has no effect, as all pairs of non-maximal edges were already identical in the corresponding triangles of the original similarity matrix A_R. A similar situation occurs in the next iteration step, where λ = 0.9 still holds as pivot weight of the edge a_{78}, and the maximal tree consists of the node set P = {7,8}. In the following iteration step, the pivot weight is lowered to λ = 0.8, whereas the set of candidate pivot edges is $\{a_{14}$,a_{27},a_{28},a_{57},$a_{58}\}$. Starting from the lexicographically first edge a_{14}, the procedure APX1A finds the maximal tree with node set P = {1,4}. Now, the local optimization step updates the weights according to the scheme

$$\begin{aligned} a_{12}, a_{21}(0.3); a_{42}, a_{24}(0.2) &\rightarrow 0.25\,, \\ a_{13}, a_{31}(0.6); a_{43}, a_{34}(0.5) &\rightarrow 0.55\,, \\ a_{15}, a_{51}(0.1); a_{45}, a_{54}(0.0) &\rightarrow 0.05\,, \\ a_{16}, a_{61}(0.6); a_{46}, a_{64}(0.5) &\rightarrow 0.55\,, \\ a_{17}, a_{71}(0.2); a_{47}, a_{74}(0.1) &\rightarrow 0.15\,, \\ a_{18}, a_{81}(0.2); a_{48}, a_{84}(0.1) &\rightarrow 0.15\,. \end{aligned}$$

Note indeed that some weights are raised while others are lowered in this stage of the algorithm. The next pivot edge a_{27}, also carrying weight λ = 0.8, gives rise to the maximal tree with node set P = {2,5,7,8}. For the first time in this example, the algorithm APX1A has encountered a maximal tree with more than two nodes, so that for all edges connecting pairs of nodes taken from the set P it needs to be checked whether they are smaller than the pivot weight. In that case, they must be increased to the level of the pivot weight. For the current example, this only happens with the weight of edge a_{25} (and its symmetric counterpart a_{52}), which is raised from the value 0.7 up to the weight 0.8 of the pivot edge. Subsequently, during the local optimization phase, the following weights are altered

$$\begin{array}{rcl} a_{21}, a_{12}(0.25); a_{51}, a_{15}(0.05); a_{71}, a_{17}(0.15); a_{81}, a_{18}(0.15) & \rightarrow & 0.15\,, \\ a_{23}, a_{32}(0.60); a_{53}, a_{35}(0.40); a_{73}, a_{37}(0.50); a_{83}, a_{38}(0.50) & \rightarrow & 0.50\,, \\ a_{24}, a_{42}(0.25); a_{54}, a_{45}(0.05); a_{74}, a_{47}(0.15); a_{84}, a_{48}(0.15) & \rightarrow & 0.15\,, \\ a_{26}, a_{62}(0.60); a_{56}, a_{65}(0.40); a_{76}, a_{67}(0.50); a_{86}, a_{68}(0.50) & \rightarrow & 0.50\,. \end{array}$$

The pivot weight then jumps to the level λ = 0.55, with the set of possible pivot edges given by {a_{13},a_{16},a_{34},a_{46}}. All these edges are interconnected at the λ-level, forming the maximal tree with node set P = {1,3,4,6}, wherein no edges need to be augmented. For the nodes external to the maximal tree, the following update scheme holds

$$\begin{array}{rcl} a_{12}, a_{21}(0.15); a_{32}, a_{23}(0.5); a_{42}, a_{24}(0.15); a_{62}, a_{26}(0.5) & \rightarrow & 0.325\,, \\ a_{15}, a_{51}(0.15); a_{35}, a_{53}(0.5); a_{45}, a_{54}(0.15); a_{65}, a_{56}(0.5) & \rightarrow & 0.325\,, \\ a_{17}, a_{71}(0.15); a_{37}, a_{73}(0.5); a_{47}, a_{74}(0.15); a_{67}, a_{76}(0.5) & \rightarrow & 0.325\,, \\ a_{18}, a_{81}(0.15); a_{38}, a_{83}(0.5); a_{48}, a_{84}(0.15); a_{68}, a_{86}(0.5) & \rightarrow & 0.325\,. \end{array}$$

Remark that for example the weight a_{12} has been iteratively updated in the last three local optimization steps. These latter changes to the weights make the matrix min-transitive, so that the remaining phases of the algorithm do not lead to any modifications of the matrix elements. The resulting tree representation is shown in dendrogram (*i*) of Figure 3.19, and the associated min-equivalence deviates 0.02790 from the original similarity matrix A_R, according to the normalized l_2-distance defined in (3.29). For the sake of completeness, in Figure 3.19 are also depicted the dendrograms associated to the min-transitive approximations of the similarity matrix A_R generated by the UPGMA algorithm (dendrogram (*ii*), $l_2(A_R, \tilde{A}_R^M) = 0.02782$), the single linkage algorithm (dendrogram (*iii*), $l_2(A_R, \tilde{A}_R^M) = 0.04831$) and the complete linkage algorithm (dendrogram (*iv*), $l_2(A_R, \tilde{A}_R^M) = 0.05440$). In this example, the two average linkage approximation algorithms UPGMA and APX1A thus score apparently better than both the min-transitive closure and the min-transitive opening according to the complete linkage method, with a slight advantage for the result of the UPGMA algorithm.

3.5.4 A second new min-transitive approximation algorithm

Looking back for a moment to the example of the trivial crisp triangular graph with weights 1, 1 and 0, if one sticks to the principle that the final weights in the graph of the min-transitive approximating fuzzy relation should be equal, as is the case for the min-transitive closure, one can assign that weight in such a manner that, according to the principle of

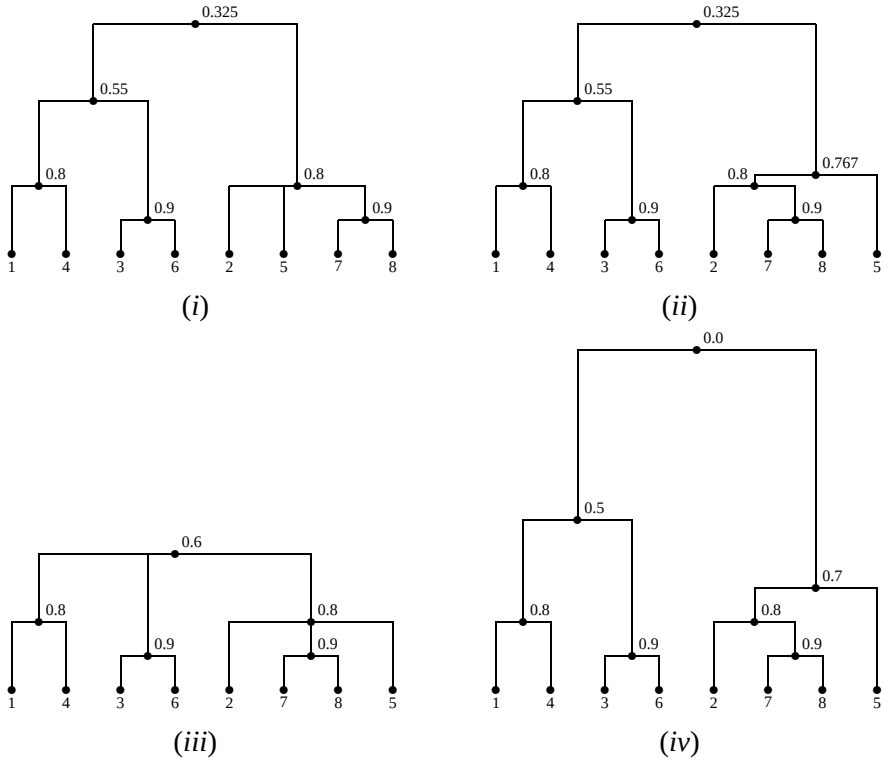


Figure 3.19: Min-transitive approximations of the similarity matrix A_R given in (3.33), according to *i*) the new algorithm APX1A, *ii*) UPGMA clustering, *iii*) single linkage clustering and *iv*) complete linkage clustering.

least squares, the changes with respect to the initial weights are minimal. Indeed, it is well-known that this can be accomplished by using the arithmetic mean of the initial weights, which in this case equals 2/3. Then, the sum of the squares of the distances between the final and initial relation is $E_2 = (1/3)^2 + (1/3)^2 + (2/3)^2 = 2/3$, which is lower than the sum for the approximation generated with algorithm APX1A, namely 1 (see Figure 3.20).

The pseudo-code procedure APX2A depicted in Figure 3.21 takes into consideration this local minimization of the squares of distances, and it is therefore expected that the min-transitive approximation generated by APX2A will, on the average, be closer to the initial similarity relation than the min-transitive approximation generated by APX1A. A price to be paid for this, is that the application of APX2A upon a crisp relation, in general no longer yields a crisp relation.

As one can verify, algorithm APX2A essentially differs from algorithm APX1A in the way the weights of the edges belonging to the subgraph with nodes in the set P (P still being the node set of a maximal tree) are updated. In APX1A all weights strictly smaller than the actual pivot value λ are raised to λ. In APX2A all weights in the subgraph that are not strictly greater than λ are averaged and raised or possibly lowered to that mean value. If none of the weights has changed, then in the subgraph there were no edges with weights smaller than λ and the algorithm continues as in APX1A. If, on the other hand, these weights have changed, then in particular the weight λ of the actual pivot edge has been lowered. This means that this edge must be temporarily abandoned as pivot edge. In

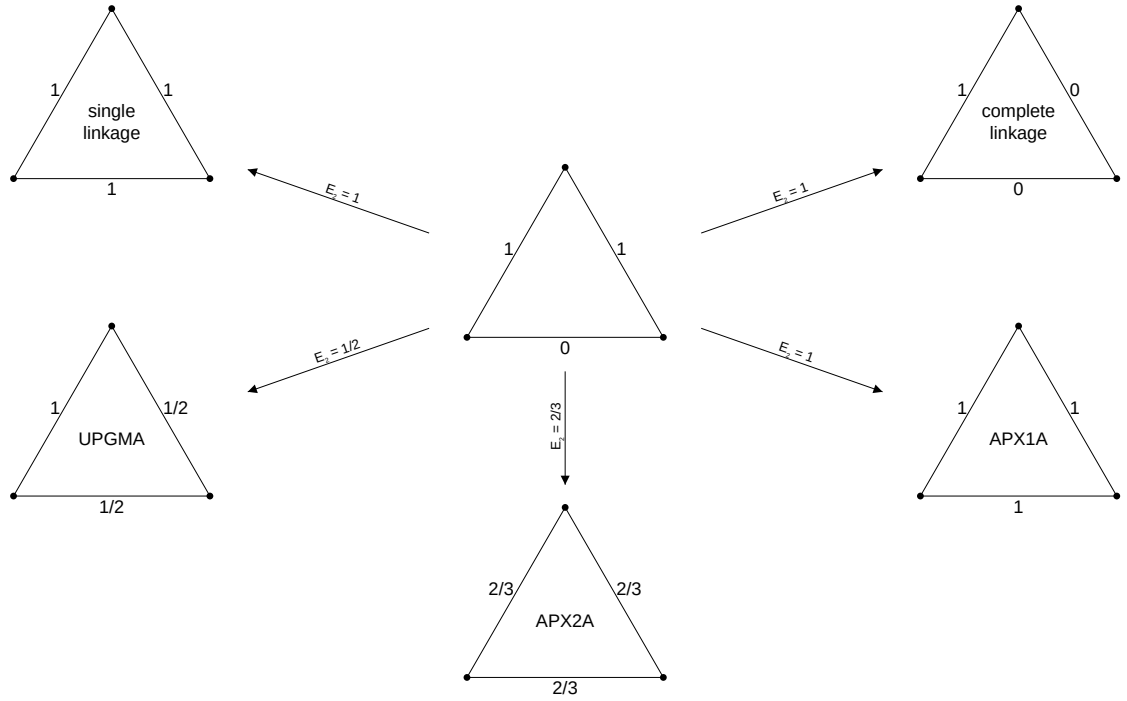


Figure 3.20: Illustration of the difference in the procedure of some min-transitive approximation algorithms on the trivial example where the graph of the given relation is a non-transitive crisp triangle with edge weights 1, 1 and 0.

a further stage of the procedure it will become pivot edge again. The validity of algorithm APX2A, namely that the final weighted graph is the graph of a min-equivalence, can be proven in the same way as for algorithm APX1A.

3.5.5 Numerical example

Let us again illustrate the new min-transitive approximation algorithm APX2A by means of the example similarity matrix A_R given in (3.33). Basically, the same scheme as outlined in subsection 3.5.3 is followed, until a_{27} is chosen as pivot edge with weight λ = 0.8. Instead of increasing the weight of the edge a_{25} (and its symmetric counterpart a_{52}) from the value 0.7 up to the weight 0.8 of the pivot edge, as done by the previous approximation algorithm APX1A, the hierarchical cluster algorithm APX2A updates all edges belonging to the complete subgraph with nodes from the maximal tree $P = \{2, 5, 7, 8\}$ that are not strictly greater than the pivot weight λ = 0.8 to the average value, in the following way

$$\begin{aligned} a_{25}, a_{52}(0.7) &\rightarrow 0.78\,, \\ a_{27}, a_{72}(0.8) &\rightarrow 0.78\,, \\ a_{28}, a_{82}(0.8) &\rightarrow 0.78\,, \\ a_{57}, a_{75}(0.8) &\rightarrow 0.78\,, \\ a_{58}, a_{85}(0.8) &\rightarrow 0.78\,. \end{aligned}$$

Remark that the edge a_{78} of the complete subgraph with nodes in P is not taken into account for this local averaging process, as it carries weight 0.9, which is above the value of the

APX2A (R, n)

Input : R, similarity relation
n, caridinality of R

Output : Min-transitive approximation $\tilde{R}$ of R, with $\tilde{R}(i,j) := w_{ij}$

begin
$S := \{1, 2, \ldots, n\}$
$V := \{w_{ij} = R(i,j) \,|\, (i,j) \in S^2\}$
$W := \{(i,j) \,|\, (i,j) \in S^2 \wedge\ i < j\}$
repeat
Select $(i,j) \in V$ such that
$(\forall\,(l,m) \in V)\ (w_{ij} \geq w_{lm})$
$\lambda := w_{ij}$
Build the maximal tree with node set P such that
$\{i,j\} \subseteq P$ and for any two adjacent nodes
$l, m \in P$ it holds that $w_{lm} = \lambda$
$q := 0\ \wedge\ c := 0$
for all $(l,m) \in P^2\ \wedge\ (l < m)$ **do**
if $w_{lm} \leq \lambda$ **then** $q := q + w_{lm} \wedge \mathrm{inc}(c)$ **endif**
endfor
$q := q/c$
if $q < \lambda$ **then**
for all $(l,m) \in P^2\ \wedge\ (l < m)$ **do**
if $w_{lm} \leq \lambda$ **then** $w_{lm} = w_{ml} := q$ **endif**
endfor
else
for all $(l,m) \in P^2\ \wedge\ (l < m)$ **do**
$V := V \setminus \{(l,m)\}$ **endfor**
for all $k \in S \setminus P$ **do**
$t := \max_{l \in P} w_{kl}$
$s := \frac{1}{\#P} \sum_{l \in P} w_{kl}$
if $t \leq \lambda$ **then**
for all $l \in P$ **do** $w_{kl} = w_{lk} := s$ **endfor**
endif
endfor
endif
until $V = \emptyset$
end

Figure 3.21: Pseudo-code of the procedure APX2A, which calculates a reflexive and symmetric min-transitive approximation and associated binary tree representation for a given similarity relation R of cardinality n.

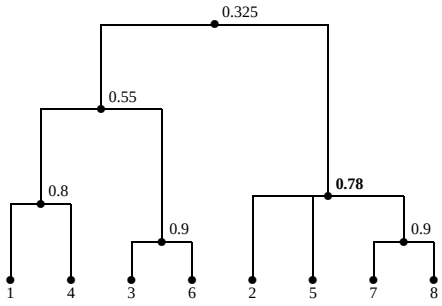


Figure 3.22: Min-transitive approximation generated by the second new algorithm APX2A for the similarity matrix A_R given in (3.33).

pivot weight. Where the procedure APX1A previously only raised weights at this stage of the algorithm, the procedure APX2A now simultaneously increases and decreases some of the weights of the edges in the complete subgraph with nodes in P. In particular, the weight of the actual pivot edge a_{27} has been lowered, so that this edge must be temporarily abandoned as pivot edge. However, for this example, the same edge a_{27} is immediately reselected as pivot edge in the next iteration step, so that the local optimization step is anyhow executed as if nothing had happened. The rest of the procedure runs unaltered in comparison to the one explained in subsection 3.5.3.

As a result, the min-equivalence generated by the algorithm APX2A has an associated dendrogram as shown in Figure 3.22. The approximative min-equivalence deviates 0.02785 from the original similarity matrix A_R, according to the normalized l_2-distance defined in (3.29). This is only slightly better than the deviation of the min-equivalence approximation produced by the procedure APX1A, as could be expected given the optimisation of the second algorithm in terms of the principle of least squares, and slightly worse than the deviation of the min-equivalence resulting from the UPGMA hierarchical clustering method. Moreover, it should be noted that the tree topology of both dendrograms generated by the new approximation algorithms is identical for the example similarity matrix A_R given in (3.33). In order to see whether a definite pattern can be recognized for the deviation caused by the min-transitive approximation algorithms touched in this section, we will carry out some more detailed experiments in the next subsection.

3.5.6 Measurement of average deviations

We want to compare the min-transitive approximations obtained with the procedures APX1A and APX2A to the approximative min-equivalences generated by calculation of the min-transitive closure, the min-transitive opening according to the complete linkage algorithm and the min-transitive approximation of the UPGMA clustering algorithm. To make relevant statistics, quite a lot of similarity relations of different cardinalities are required. Also, one needs to consider sufficient variations between the extreme cases of all initial weights equal and all initial weights different. Clearly, real data are lacking to carry out such reliable tests and one needs to construct synthetic data that simulate as much as pos-

sible the similarity matrices arising from real experiments.

We recall from our detailed investigation of the min-transitive opening deviations in subsection 3.4.6, that random matrices are not the kind of matrices that stand model for the similarity matrices encountered in practical applications. Therefore, we will restrict ourselves here to randomly generated matrices of type-2 for performing a statistical analysis on the deviations of min-equivalences generated by different approximation algorithms. The precision of these matrices is still indicated by the integer value N. We have carried out tests to compare the results of five approximative min-equivalence procedures: *i*) the first new approximation algorithm APX1A where within-cluster weights are set to the pivot weight, *ii*) the second new approximation algorithm APX2A where within-cluster weights are averaged, *iii*) the UPGMA algorithm, *iv*) the single linkage clustering algorithm (SL) which generates the min-transitive closure and *c*) the complete linkage clustering algorithm (CL) which generates a representative min-transitive opening. We made the distinction between two test sets of random matrices: a set of matrices of precision $N = 1$ (initial weights can take 11 possible values) and a set of matrices of precision $N = 2$ (initial weights can take 101 possible values). Each of these test sets contains matrices of which the dimension n is a tenfold situated between 10 and 100. For each dimension n and precision N, 1000 sample type-2 matrices have been generated. The distance between a min-transitive approximation matrix and the initial matrix was computed using the normalized l_2-distance defined for reflexive and symmetric matrices in (3.29). Notice however that the choice between the normalized l_1-distance and l_2-distance does not drastically influences the outcome of the statistical analysis.

In Figures 3.23 and 3.24, against the dimension n of the type-2 input matrices A_{t_2} are plotted the average distances of the five approximation matrices $\tilde{A}_{t_2}$ (denoted SL, CL, UPGMA, APX1A and APX2A) with respect to an initial Łukasiewicz-transitive similarity matrix. Figure 3.23 shows the results for the case of matrix precision $N = 1$ (initial matrix elements rounded up to first decimal), whereas Figure 3.24 is concerned with matrices of precision $N = 2$ (initial matrix elements rounded up to the second decimal). It should be noted that this fixed precision is only a property of the input matrices, since all further computations are performed with machine accuracy. It nevertheless accounts for the number of times on the average a same initial weight occurs in the graph of the input fuzzy relation.

It is immediately apparent from both figures that the newly generated approximations, as well as the UPGMA-approximation, are on the average significantly closer to the initial matrix than are the min-transitive closure and the min-transitive opening resulting from the complete linkage clustering. This is the major explanation of the fact that the application of average linkage clustering algorithms is far more widespread in the research area of microbial taxonomy, than is the utilization of single linkage and complete linkage clustering. Furthermore, as far as the average distance is concerned, we see that the differences between our new approximations and the UPGMA-approximation become obsolete once the initial matrix elements are almost all different. This makes Figure 3.24 difficult to read, as the curves labelled APX1A, APX2A and UPGMA are nearly completely coincident.

With the philosophy of the new min-transitive approximation procedures in mind, it

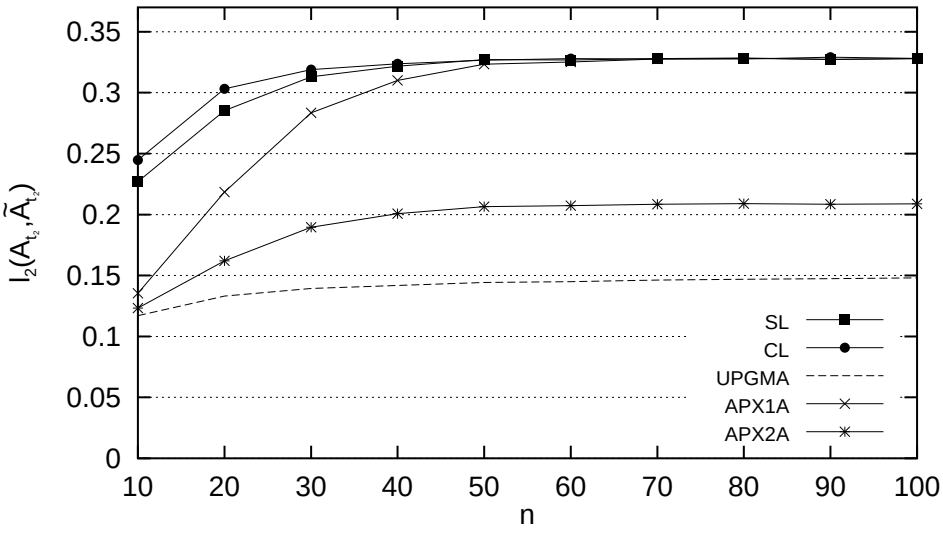


Figure 3.23: Average l_2-distances of five types of approximation matrices to an initial type-2 similarity matrix of dimension n and of precision $N = 1$.

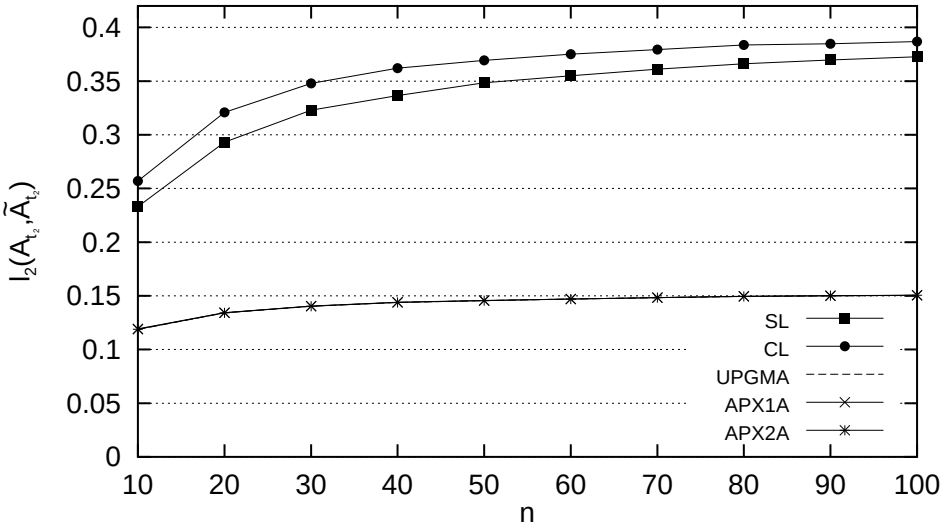


Figure 3.24: Average l_2-distances of five types of approximation matrices to an initial type-2 similarity matrix of dimension n and of precision $N = 2$.

is not surprising to see that the major differences between these algorithms, as far as the average distance is concerned, are most prominent when the number of equal elements within the initial similarity matrices is high, as is definitely the case for the larger matrix dimensions in Figure 3.23. Looking more closely at the curves in this figure, we see that the approximation algorithm APX1A seems to degenerate completely to the level of the min-transitive closure for initial matrices with many ties among the matrix elements, whereas the algorithm APX2A yields approximations that strikingly well keep pace with the approximations generated by the UPGMA algorithm, notwithstanding the extra level of objectivity built into its procedure. These observations are totally in agreement with those found for the trivial example of Figure 3.20, which means that as far as the average distance is concerned, the local optimization of distances between matrix elements also favours the global optimization of distances. Finally, since algorithm APX1A requires less computational efforts than algorithm APX2A, both in the instructional descriptions given here as well as in their optimal implementation, we can conclude that algorithm APX1A should be preferred for general situations, unless the input relation is crisp or has nearly all its elements equal.

3.5.7 Min-transitive approximations using median linkage

Throughout this section, calculation of weight averages has been consistently used as the agglomerative operation of choice for the linkage of clusters, because of its intrinsic qualities in minimising the error sum of squares E_2. However, a criticism of this option could be that it leads to unnecessary intermediate levels that were not present in the original similarity model, or stated differently, that the average linkage methods get a large part of their benefit with respect to the min-transitive closure and opening algorithms, by affecting the initial accuracy of the similarity matrices.

To avoid this behaviour, it is possible to replace calculations of weight averages in the new approximation algorithms, by the value of the median among the corresponding group of weights. As a result, the first new approximation algorithm can be implemented according to the pseudo-code procedure APX1M shown in Figure 3.25, while the pseudo-code procedure APX2M of Figure 3.26 is similarly derived from the second new approximation algorithm. Remark that for the latter algorithm, both average weight calculations have been replaced by the derivation of the median of the weights. Note also that the last letter of the name of the new approximation procedures indicates whether average linkage (A) or median linkage (M) was used in the agglomerative measure when merging clusters. In case the median value needs to be determined from an even set of weights, the median linkage algorithms choose the maximum of the two middle weights, keeping in mind that the new algorithms are derived from an algorithm for calculating the min-transitive closure.

When applied upon the similarity matrix A_R given in (3.33), both new median linkage approximation algorithms APX1M and APX2M result in the same min-equivalence, having an associated binary matrix representation as depicted in Figure 3.27. The approximative min-equivalence deviates 0.03763 from the initial similarity matrix A_R, according

APX1M (R, n)

Input : R, similarity relation
n, caridinality of R

Output : Min-transitive approximation $\tilde{R}$ of R, with $\tilde{R}(i,j) := w_{ij}$

begin
$S := \{1, 2, \ldots, n\}$
$V := \{w_{ij} = R(i,j) \,|\, (i,j) \in S^2\}$
$W := \{(i,j) \,|\, (i,j) \in S^2 \wedge i < j\}$
repeat
Select $(i,j) \in V$ such that
$(\forall\, (l,m) \in V)\ (w_{ij} \geq w_{lm})$
$\lambda := w_{ij}$
Build the maximal tree with node set P such that
$\{i,j\} \subseteq P$ and for any two adjacent nodes
$l, m \in P$ it holds that $w_{lm} = \lambda$
for all $(l,m) \in P^2 \wedge (l < m)$ **do**
$V := V \setminus \{(l,m)\}$
if $w_{lm} < \lambda$ **then** $w_{lm} = w_{ml} := \lambda$ **endif**
endfor
for all $k \in S \setminus P$ **do**
$t := \max_{l \in P} w_{kl}$
if $t \leq \lambda$ **then**
$c := 0$
for all $l \in P$ **do** $inc(c) \wedge K[c] := w_{kl}$ **endfor**
Sort the values in K in decreasing order
$s := K[\lfloor \frac{c-1}{2} \rfloor + 1]$
for all $l \in P$ **do** $w_{kl} = w_{lk} := s$ **endfor**
endif
endfor
until $V = \emptyset$
end

Figure 3.25: Pseudo-code of the procedure APX1M, which calculates a reflexive and symmetric min-transitive approximation and associated binary tree representation for a given similarity relation R of cardinality n.

APX2M (R, n)

Input : R, similarity relation
n, caridinality of R

Output : Min-transitive approximation $\tilde{R}$ of R, with $\tilde{R}(i,j) := w_{ij}$

begin
$S := \{1, 2, \ldots, n\}$
$V := \{w_{ij} = R(i,j) \,|\, (i,j) \in S^2\}$
$W := \{(i,j) \,|\, (i,j) \in S^2 \wedge i < j\}$
repeat
Select $(i,j) \in V$ such that
$(\forall\,(l,m) \in V)\ (w_{ij} \geq w_{lm})$
$\lambda := w_{ij}$
Build the maximal tree with node set P such that
$\{i,j\} \subseteq P$ and for any two adjacent nodes
$l, m \in P$ it holds that $w_{lm} = \lambda$
$c := 0 \wedge t := 1$
for all $(l,m) \in P^2 \ \wedge\ (l < m)$ **do**
if $w_{lm} \leq \lambda$ **then** $inc(c) \wedge K[c] := w_{lm} \wedge t := \min(t, w_{lm})$ **endif**
endfor
if $t < \lambda$ **then**
Sort the values in K in decreasing order
$q := K[\lfloor \frac{c-1}{2} \rfloor + 1]$
for all $(l,m) \in P^2 \ \wedge\ (l < m)$ **do**
if $w_{lm} \leq \lambda$ **then** $w_{lm} = w_{ml} := q$ **endif**
endfor
else
for all $(l,m) \in P^2 \ \wedge\ (l < m)$ **do**
$V := V \setminus \{(l,m)\}$ **endfor**
for all $k \in S \setminus P$ **do**
$t := \max_{l \in P} w_{kl}$
if $t \leq \lambda$ **then**
$c := 0$
for all $l \in P$ **do** $inc(c) \wedge K[c] := w_{kl}$ **endfor**
Sort the values in K in decreasing order
$s := K[\lfloor \frac{c-1}{2} \rfloor + 1]$
for all $l \in P$ **do** $w_{kl} = w_{lk} := s$ **endfor**
endif
endfor
endif
until $V = \emptyset$
end

Figure 3.26: Pseudo-code of the procedure APX2M, which calculates a reflexive and symmetric min-transitive approximation and associated binary tree representation for a given similarity relation R of cardinality n.

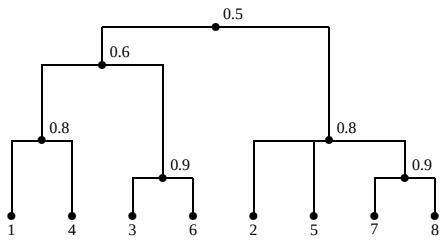


Figure 3.27: Dendrogram associated to the min-transitive approximation generated by both the new median linkage algorithms `APX1M` and `APX2M`, for the example similarity matrix A_R given in (3.33).

to the normalized l_2-distance defined in (3.29). Given the additional restrictions on the resulting min-equivalence, it is not unexpected that this min-transitive similarity matrix deviates somewhat more from the initial similarity matrix than are the min-equivalences attained by the average linkage methods. However, the median methods still produce a min-equivalence that is significantly closer to the original matrix than are the single linkage and complete linkage clusterings. All new approximation algorithms result in min-equivalences with the same tree topology for this example.

We have repeated the same statistical measurements for the average deviations of the median linkage cluster methods, as in our setup for the average linkage clustering methods (see subsection 3.5.6). Figure 3.28 shows the deviations with increasing type-2 matrix dimensions n for the case of matrix precision $N = 1$ (initial matrix elements rounded up to first decimal), whereas Figure 3.29 is concerned with type-2 matrices of precision $N = 2$ (initial matrix elements rounded up to the second decimal). The same general conclusions can be drawn for the deviations attained with the new median linkage approximations in comparison to the single linkage, complete linkage and UPGMA cluster algorithms, as for the new average linkage algorithms. Note however that the extra limitations on the accuracy of the matrix elements for the final min-transitive approximations, have indeed led to that fact that the new median linkage methods have somewhat floated away from the average linkage methods, as far as distance to the initial matrix is concerned, but still score on average significantly better than both the single linkage and complete linkage results.

3.6 Conclusions and future perspectives

In this chapter, we have simplified the complete linkage clustering algorithm in such a way that in realistic situations it generates a reflexive and symmetric min-transitive opening of a similarity matrix that is closer to the initial matrix than with other opening algorithms and even runs faster than these algorithms. We have also established two new weight-driven algorithms that, for generating a min-transitive approximation of a similarity relation, are as efficient as the UPGMA clustering algorithm and yield approximations that are comparably close to the given relation. Unlike the UPGMA algorithm, the new algorithms do not directly rely on the aggregation of clusters, although the computational steps depend upon the descending order in which weights are selected in order to locally impose the property

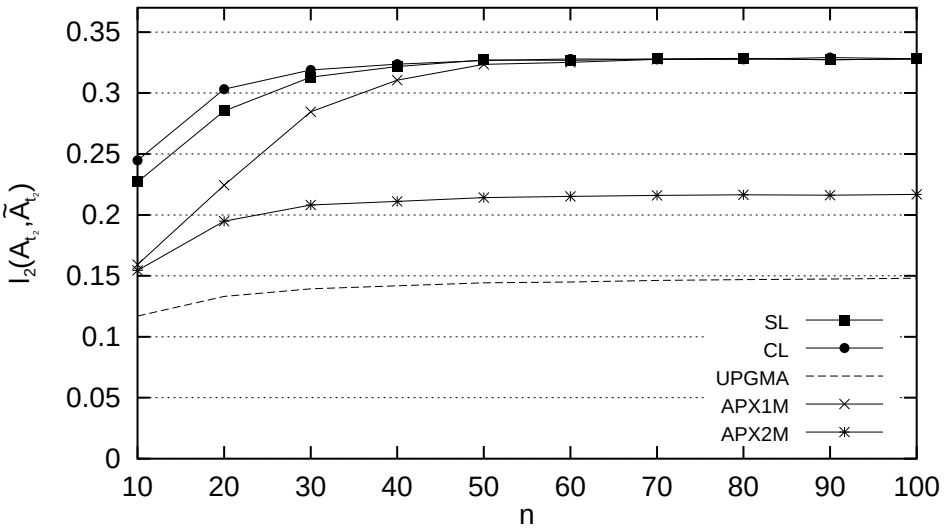


Figure 3.28: Average l_2-distances of five types of approximation matrices to an initial similarity matrix of dimension n and of precision $N = 1$.

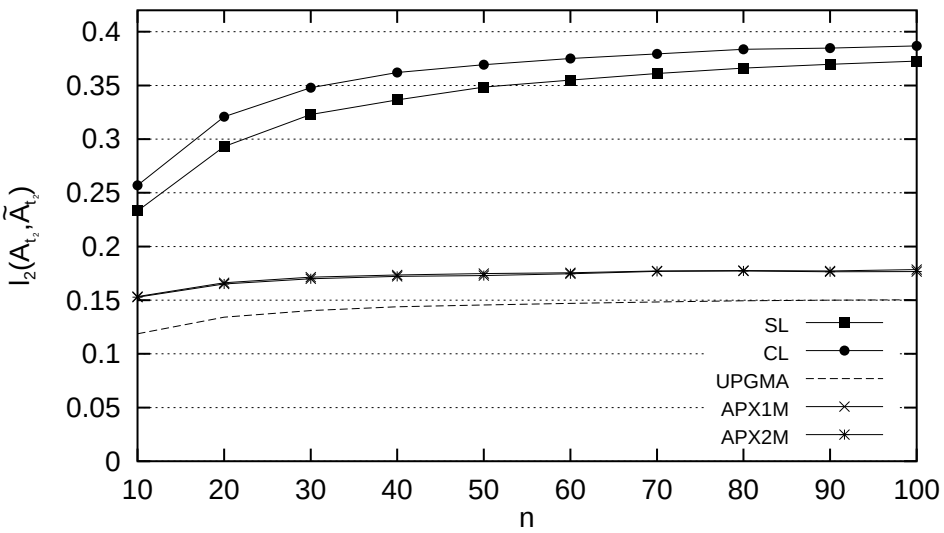


Figure 3.29: Average l_2-distances of five types of approximation matrices to an initial similarity matrix of dimension n and of precision $N = 2$.

of min-transitivity. This approach has the advantage that, at least in principle, it opens possibilities for modifying these algorithms in two different ways, namely to establish algorithms that generate a T-transitive approximation of a given similarity relation, and to establish algorithms that generate a min-transitive approximation (a fuzzy preorder) of a given reflexive fuzzy relation. Note that no variants of the UPGMA clustering algorithm are known that are applicable for non-symmetric fuzzy relations.

A choice between this wealth of min-transitive approximation techniques might be driven by a series of evaluation criteria that were pointed out during this chapter. First of all, we have used the error sum of squares (or an equivalent distance measure) between the approximated min-equivalences and their original similarity matrices, as a means to estimate the overall impact of the approximation algorithms on the genuine similarity model. Furthermore, some methods suffer from subjective decision making during their execution. As a result, the generated min-transitive approximations are not invariant under permutation of the objects. The time and space complexity of the algorithms can be equally important for the classification of large numbers of objects. Finally, some algorithms restrict the search space of surrounding min-equivalences by the fact that they do not introduce new fuzzy weight values, or stated differently, they do not affect the accuracy of the origal similarity model. Other methods conversely tend to introduce intermediate levels of fuzzy values, which leads in the special case of crisp similarity matrices to a fuzzification of the equivalence model.

Bibliography

[1] **Anderberg, M. R. (1973).** Cluster analysis for applications. Academic Press, New York and London, NY, USA.

[2] **Bandler, W. & Kohout, L. (1988).** Special properties, closures and interiors of crisp and fuzzy relations. *Fuzzy Sets and Systems* **26**, 317–331.

[3] **Boixader, D. (1998).** Some properties concerning the quasi-inverse of a t-norm. *Mathware & Soft Computing* **5**, 5–12.

[4] **Darwin, C. (1859).** On the Origin of Species by Means of Natural Selection, or, The Preservation of Favoured Races in the Struggle for Life. John Murray publishers of London.

[5] **Dawyndt, P, De Meyer, H., De Baets, B. & Swings, J. (2002).** A fast algorithm for generating a min-transitive opening of a similarity relation. In: *Proceedings of the EUROFUSE Workshop on Information Systems*, Villa Monastero, Varenna, Italy, September 23-25, 2002.

[6] **Dawyndt, P., De Meyer, H. & De Baets, B. (in press).** The complete linkage clustering algorithm revisited. *Soft Computing*. DOI: 10.1007/s00500-003-0346-3.

[7] **Dawyndt, P, De Meyer, H. & De Baets, B. (2004).** On the min-transitive approximation of symmetric fuzzy relations. In: *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2004)*, Budapest, Hungary, 25-29 July, 2004.

[8] **Dawyndt, P., Thompson, F.L., Austin, B., Swings, J., Koski, T. & Gyllenberg, M. (in press).** Application of sliding window discretization and minimization of stochastic complexity for the analysis of FAFLP genotyping fingerprint patterns of *Vibrionaceae*. *International Journal of Systematic and Evolutionary Microbiology*.

[9] **De Baets, B., De Meyer, H. & Naessens, H. (2001).** A class of rational cardinality-based similarity measures. *Journal of Computational and Applied Mathematics* **132(1)**, 51–69.

[10] **De Baets, B. & De Meyer, H. (2003).** Transitive approximation of fuzzy relations by alternating closures and openings. *Soft Computing* **7**, 210–219.

[11] **De Baets, B. & De Meyer, H. (2003).** On the existence and construction of T-transitive closures. *Information Sciences* **152**, 167–179.

[12] **De Baets, B., De Meyer, H. & Naessens, H. (in press).** A top-down algorithm for generating the Hasse tree representation of the preorder closure of a fuzzy relation. *IEEE Transactions on Fuzzy Systems*.

[13] **De Meyer, H., Naessens, H. & De Baets, B. (2004).** Algorithms for computing the min-transitive closure and associated partition tree of a symmetric fuzzy relation. *European Journal of Operational Research* **155**, 226–238.

[14] **Devillez, A., Billaudel, P. & Lecolier, G. (2002).** A fuzzy hybrid hierarchical clustering method with a new criterion able to find the optimal partition. *Fuzzy Sets and Systems* **128**, 323–338.

[15] **Dice, L. R. (1945).** Measures of the amount of ecological association between species. *Ecology* **26**, 297–302.

[16] **Dunn, J. (1974).** A graph-theoretical analysis of pattern classification via Tamura's fuzzy relation. *IEEE Transactions on Systems, Man and Cybernetics* **5**, 310–313.

[17] **Feijs, L. & van Ommering, R. (1997).** Abstract derivation of transitive closure algorithms. *Information Processing Letters* **63**, 159–164.

[18] **Fodor, J. & Roubens, M. (1995).** Structure of transitive valued binary relations. *Mathematical Social Sciences* **30**, 71–94.

[19] **Fu, G. (1992).** An algorithm for computing the transitive closure of a fuzzy similarity matrix. *Fuzzy Sets and Systems* **51**, 189–194.

[20] **Garrity, G. M., Johnson, K. L., Bell, J. A. & Searles, D. B. (2002).** Taxonomic Outline of the Procaryotes. In: *Bergey's Manual of Systematic Bacteriology*, 2nd Edition, Release 3.0, Springer–Verlag, New York, NY, USA. DOI:10.1007/bergeysoutline200210.

[21] **Gyllenberg, M., Koski T. & Verlaan, M. (1997).** Classification of binary vectors by stochastic complexity. *J Multivariate Anal* **63**, 47–72.

[22] **Gyllenberg, M., Koski, T., Dawyndt, P., Lund, T., Thompson, F., Austin, B. & Swings, J. (2002).** New methods for the analysis of binarized BIOLOG GN data of Vibrio species: minimization of stochastic complexity and cumulative classification. *Systematic and Applied Microbiology* **25**, 403–415.

[23] **Heuchenne, C. (1969).** Sous-relations transitives maxima. *Bulletin de la Société Royale des Sciences de Liège* **38**, 435–449.

[24] **Jaccard, P. (1908).** Nouvelles recherches sur la distribution florale. *Bulletin de la société Vaudoise des Sciences Naturelles* **44**, 223–270.

[25] **Kandel, A. & Yelowitz, L. (1974).** Fuzzy chains. *IEEE Transactions on Systems, Man and Cybernetics* **5**, 472–475.

[26] **Klement, E., Mesiar, R. & Pap, E. (2000).** Triangular Norms. In: *Trends in Logic*, Studia Logica Library, Vol. 8, Kluwer Academic Publishers, Dordrecht.

[27] **Kundu, S. (2000).** An optimal $\mathcal{O}(N^2)$ algorithm for computing the min-transitive closure of a weighted graph. *Information Processing Letters* **74**, 215–220.

[28] **Larsen, H. & Yager, R. (1990).** Efficient computation of transitive closures. *Fuzzy Sets and Systems* **38**, 81–90.

[29] **Leclerc, B. (1986).** Caractérisation, construction et dénombrement des ultramétriques supérieures minimales. *Statistique et Analyse des Données* **11**, 26–50.

[30] **Lee, H. (2001).** An optimal algorithm for computing the max-min transitive closure of a fuzzy similarity matrix. *Fuzzy Sets and Systems* **123**, 129–136.

[31] **Li, S.-Y. (1990).** The simplest method of ascending value to find fuzzy transitive closure. *Fuzzy Sets and Systems* **38**, 91–96.

[32] **Miller, R. & Boxer L. (2000).** Algorithms Sequential and Parallel, a Unified Approach. Prentice Hall, Upper Saddle River, NJ, USA.

[33] **Naessens, H., De Meyer, H. & De Baets, B. (2002).** Algorithms for the computation of T-transitive closures. *IEEE Transactions on Fuzzy Systems* **10(4)**, 541–551.

[34] **Saitou, N. & Nei, M. (1987).** The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular Bioliology and Evolution* **4(4)**, 406–425.

[35] **Sokal, R. & Michener, C. D. (1958).** A statistical method for evaluating systematic relationships. *University of Kansas Scientific Bulletin* **38**, 1409–1438.

[36] **Sneath, P. & Sokal, R. (1973).** Numerical Taxonomy. Freeman, San Fransisco, USA.

[37] **Schweizer, B. & Sklar, A. (1983).** Probabilistic Metric Spaces. Elsevier North Holland, New York, NY, USA.

[38] **Ullman, J.& Yannakis, M. (1990).** The input/output complexity of transitive closure. In: *Proceedings of the ACM-SIGMOD 1990 International Conference on Management of Data*, 44–53.

[39] **Vandamme, P., Pot, B., Gillis, M., De Vos, P., Kersters, K. & Swings, J. (1996).** Polyphasic taxonomy, a consensus approach to bacterial systematics. *Microbiological Review* **60**, 407–438.

[40] **Yeung, D. & Wang, X. (2002).** Improving performance of similarity-based clustering by feature weight learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**, 556-561.

[41] **Vancanneyt, M., Mengaud, J., Cleenwerck, I., Vanhonacker, K., Hoste, B., Dawyndt, P., Degivry, M. C., Ringuet, D., Janssens, D., Swings, J. (2004).** Reclassification of *Lactobacillus kefirgranum* Takizawa et al. 1994 as *Lactobacillus kefiranofaciens* subsp. *kefirgranum* subsp. nov. and emended description of *L. kefiranofaciens* Fujisawa et al. 1988. *Int J Syst Evol Microbiol* **54(2)**, 551–556.

[42] **Ward, J. H. (1963).** Hierarchical grouping to optimize an objective function. *Journal of American Statistical Association* **58**, 236–244.

[43] **Warshall, S. (1962).** A theorem on boolean matrices. *Journal of the ACM* **9(1)**, 11–12.

[44] **Zadeh, L. (1971).** Similarity relations and fuzzy orderings. *Information Sciences* **3, 177–200.**

Chapter 4

The content of this chapter is a strongly based on the published or submitted material in the following scientific journal papers:

[1] **Gyllenberg, M., Koski, T., Dawyndt, P., Lund, T., Thompson, F., Austin, B. & Swings, J. (2002).** New methods for the analysis of binarized BIOLOG GN data of *Vibrio* species: minimization of stochastic complexity and cumulative classification. *Syst Appl Microbiol* **25(3)**, 403–415.

[2] **Austin, B., Dawyndt, P., Gyllenberg, M., Koski, T., Lund, T., Swings, J. & Thompson, F. L. (2004).** Sliding window discretization: a new method for multiple band matching of bacterial genotyping fingerprints. *Bull Math Biol* **66(6)**, 1575–1596.

[3] **Dawyndt, P., Thompson, F. L., Austin, B., Swings, J., Koski, T. & Gyllenberg, M. (in press).** Application of sliding window discretization and minimization of stochastic complexity for the analysis of fAFLP genotyping fingerprint patterns of *Vibrionaceae*. *Int J Syst Evol Microbiol*. DOI:10.1099/ijs.0.63136-0.

Chapter 4

Application of Sliding Window Discretization and Minimization of Stochastic Complexity for the Analysis of Bacterial Genotyping Fingerprints

"Science is an endless search for truth. Any representation of reality we develop can be only partial. There is no finality, sometimes no single best representation. There is only deeper understanding, more revealing and enveloping representations."

— Carl R. Woese

EVER since the pioneering developments of the computational techniques used in the field of numerical taxonomy in the 1950's, microbiologists have traditionally applied hierarchical clustering algorithms as their mathematical tool of choice to unravel the taxonomic relationships between microorganisms. However, this total reliance on the interpretation of such stratified classifications suffers from being subjective, in that a wide variety of *ad hoc* choices must be made during their construction. On the other hand, the employment of more profound and objective mathematical methods – such as the minimization of stochastic complexity – for the classification of bacterial genotyping fingerprint data is seriously hampered by the prerequisite that this kind of methods only acts upon vectorized data representations.

In this chapter, we have sought to shatter this tradition by adopting minimization of stochastic complexity (SC) as a new strategy for the classification of genotyping fingerprinting profiles. Because the current BinClass implementation of this iterative optimization algorithm for classification only works upon binary feature vectors, a new technique,

coined sliding window discretization, for the transformation of genotypic fingerprint patterns into binary vector format is presented. Within the context of an extensive fluorescent amplified fragment length polymorphism (fAFLP) data set of 507 strains from the *Vibrionaceae* family, that has been analysed before following a traditional approach, we demonstrate that the new sliding window discretization results in minimal loss of the original information content captured in the banding patterns, in comparison to a number of alternative discretization techniques. This new multiple band matching algorithm althus extends the microbiologist's toolbox of data mining techniques, by enabling the application of state-of-the-art non-hierarchical classification methods for learning the hidden bacterial relationships behind sets of genotypic fingerprint patterns.

The novel classification generated using the BinClass software package has been subjected to an in–depth comparison with the hierarchical classification of the same data set, in order to acknowledge the applicability of the new classification strategy as a more objective approach for the classification of genotyping fingerprint patterns. Recent DNA–DNA hybridization and 16S rRNA sequence experiments proved that the classification based on SC-minimization forms separate clusters that contain the fAFLP patterns for all representatives of the species *Enterovibrio norgevicus*, *Vibrio fortis*, *Vibrio diazothropicus* and *Vibrio campbellii*, whereas previous hierarchical cluster analysis had suggested more heterogeneity within the fAFLP patterns by splitting the representatives of the above-mentioned species into multiple distant clusters. As a result, the new classification methodology has highlighted some previously unseen relationships within the biodiversity of the family *Vibrionaceae*. This new taxonomical knowledge learned from an alternative classification of the fAFLP patterns endorses the value of combining sliding window discretization with minimization of stochastic complexity, as a clarifying classification strategy for the analysis of bacterial genotyping fingerprints.

4.1 Introduction

Bacterial taxonomy and phylogeny have a long standing tradition of relying on a small range of hierarchical clustering algorithms to establish taxonomies based on the phenotypic and genotypic characteristics of microorganisms, with the output being represented in the form of similarity matrices and dendrograms/trees. Notwithstanding, there has been some concern about the subjective nature of the process [80]. It is generally appreciated that when using a hierarchical method of cluster analysis, early decisions in the construction process may preclude certain meaningful groupings at later stages [97]. Moreover, for a given data set there might be many meaningful groupings, that each reflect different aspects of the underlying relationships. Therefore, a single classification may give a distorted view of the multifaceted set of patterns. Consequently, if there are several meaningful groupings, a variety of cluster analysis techniques will be needed to reveal them all [2].

Despite new developments in microbial taxonomy towards the usage of genotypic information, there has thus been a tendency to rely on traditional mathematical methods for the data analysis [80]. This is regrettable because, as a consequence, biology does not benefit

from the modern developments in mathematics and computational techniques. Standard hierarchical clustering algorithms are still commonly used for classification of molecular fingerprints, as their similarity model is based on the straightforward calculation of an intermediate pairwise similarity or dissimilarity matrix [2, 97]. However, the weakness of this family of methods lies in its overwhelming variety of *ad hoc* choices that must be made (e.g. the choice of a band matching algorithm, a similarity measure, a clustering method, an optimal cluster-cutoff level), which leaves much of the subjective decision making up to the taxonomist.

Inspired by these observations, the family of well-founded mathematical classification methods that are based on the optimization of an information theoretic expression such as entropy [39] or stochastic complexity [41], offers the opportunity to become a valuable complementary alternative for the hierarchical classifications used in bacterial taxonomy [36, 37, 38, 43]. However, where hierarchical clustering methods work upon a matrix of pairwise determined similarity or dissimilarity values as to which they can make use of both pairwise and multiple band matching methods to compare the fingerprint patterns, these optimization methods are designed to work directly on vectorized data sets. Application of this kind of methods for the classification and validation of molecular fingerprint profiles is thus hindered by the required additional transformation of the banding patterns into (binary) vector format. We will demonstrate that several existing discretization methods may lead to an unacceptable reduction of the information content stored in the original fingerprint patterns of some data sets, which has a dramatic negative impact on the final classification results.

Within the framework of this study we present a new method, termed *sliding window discretization*, for the transformation of molecular banding profiles, as obtained in bacterial genotyping, into binary vector format. Clearly these transformations need to be performed with minimal loss of information, in order to minimize their effect on further stages of the data analysis. We will demonstrate in the context of a large data set of fluorescent amplified fragment length polymorphism (fAFLP, [54]) fingerprint patterns of strains from the family *Vibrionaceae*, that this sliding window method conducts the transformation with better conservation of the original information content in respect to other transformation methods. This will ultimately have a beneficial effect on classifications calculated from a vector representation of the genotypic fingerprints within this data set. The sliding window discretization technique is introduced in this chapter especially for application on the classification of genotyping fingerprint patterns of bacterial strains. However, we want to stress here that the method might be more generally applicable into any problem domain that has a need for transforming continuous data sets into a discrete representation, in most cases as a preliminary step for further data analysis.

The current chapter sets off with a brief description on the nature of genotypic fingerprint patterns (section 4.2) and alternative approaches to compare these molecular signatures by matching their densitometric curves or common bands (section 4.3). Most taxonomic publications that build on the classification of genotypic fingerprint patterns tend to give little or no information on the way different banding profiles are matched with each other. Therefore, we have devoted two sections of this chapter at establishing a classification of

the different band matching algorithms found in the literature or in special-purpose software packages designed for clustering molecular fingerprints, where we introduce some new terminology and give a formal discussion on each of the band matching algorithms. Section 4.6 reviews several pairwise band matching algorithms, while section 4.7 discusses some multiple band matching algorithms that belong to the family of methods for transforming band patterns into binary vector format. Although a member of the same family, the new sliding window band matching algorithm is introduced separately in section 4.8, where we give a formal definition of the method and demonstrate its procedure onto a simple artificial example. In section 4.10, we present an overview on the application of minimization of stochastic complexity for the classification of binary feature vectors and its implementation in the BinClass software package.

The case study dealt with in section 4.11 is conceived as a proof of principle, where the scientific value of sliding window discretization is proven in the framework of an extended data set of fluorescent amplified fragment length polymorphism fingerprint patterns of *Vibrionaceae* strains. Thompson *et al.* [103] have analyzed this set of 507 fAFLP fingerprint profiles based on a hierarchical classification using Ward's hierarchical clustering algorithm [119]. This algorithm does not make a direct classification of the fingerprint patterns, but rather works upon an intermediate similarity or dissimilarity matrix. In their paper, Thompson *et al.* [103] calculated this similarity matrix using the Dice similarity coefficient s_D [25]. From the hierarchical clustering, a plain classification was derived by selecting a rather arbitrary α–cut based on the intuition of the authors and the distribution of type and reference strains included in the data set. We will show for this set of data that the sliding window method scores higher compared to other discretization methods, in that it leads to a better conservation of the information content of the original fingerprint patterns after discrete transformation. Based on this observation, the new discretization method was chosen as data preprocessor for reclassification of the same set of fingerprint patterns based on more objective foundations and without taking into account prior knowledge about the group of bacterial strains. To this means, the classification method based on minimizing the stochastic complexity of a binary vector representation of the data [40] was chosen as a representative method from the family of evaluation function optimization algorithms for classification. This method has been implemented in the BinClass software package [42], which does not not only enable the calculation of an optimal classification in the sense of stochastic complexity, but also allows the construction of a stochastic complexity-driven hierarchy built on top of the optimal classification [37]. Both classifications have been subjected to a profound comparison, which shows a global correspondence between the major parts of the two groupings. However, differences found between the classifications have stimulated the discovery of new relationships within the taxonomy of the *Vibrionaceae* strains, that have been confirmed by recent DNA–DNA hybridization and 16S rDNA sequence experiments [105, 106, 107, 108]. These results prove the value of the new methodology as an alternative classification strategy in its own right, and the general need *i*) to inspect a given data set from different angles using different mathematical models, in order to get a complete picture of all the relationships present in the data, and *ii*) to test the robustness of the different classifications by investigation of their concordances and disparities, in an attempt to sort out the artefacts inherent to the usage of the classification procedures.

4.2 Genotypic fingerprinting techniques

Over the past two decades, molecular biologists have developed a tremendous variety of tools and techniques that directly reflect the whole or parts of the bacterial genome. This has led to the construction of bacterial taxonomies based on detection of the naturally occurring DNA polymorphisms [30, 74]. These evolutionary polymorphisms are the result of naturally occurring point mutations (insertions, deletions, substitutions and inversions) or large scale genetic rearrangements (gene duplications and transpositions) in the DNA, and can be detected by scoring band presence versus absence in the genotypic fingerprinting patterns that are generated by, e.g., DNA amplification procedures [49, 54]. In principle, a restriction endonuclease (or restriction enzyme) recognizes a specific sequence of nucleotide pairs and cleaves it. These enzymes are produced naturally by bacteria as a mechanism for attacking foreign DNA from viruses, called bacteriophages, that intend to penetrate the bacterial DNA. This desintegration process is called a *digest*, and the combined application of multiple enzymes creates a collection of restriction fragments, which are cut up pieces of the original microbial DNA. The number and locations of restriction sites vary with nucleotide sequence.

The *polymerase chain reaction* (PCR, Figure 4.1), though strikingly simple in both its theory and practice, is a very powerful mechanism that makes it possible to rapidly produce huge amounts of a specific region of DNA, simply by knowledge of a little bit of the sequence around the desired region. PCR exponentially amplifies (makes copies of) the target DNA sequence, given a unique pair of sequences that bracket the desired piece. First, short sequences of DNA (called *oligonucleotides*, or *oligos* for short) complementary to each of the bracketing sequences are synthesized. Creating short pieces of DNA with a specific sequence is routine technology, nowadays often performed by laboratory robots. These pieces are called *primers*. The primers, the target DNA and the enzyme DNA polymerase are then combined. This mixture is heated, so that the hydrogen bonds in the DNA break and the two strands of the double helix are separated, a process commonly called *denaturation*. When the mixture cools sufficiently, the primers bind to the regions around the area of interest, and the DNA polymerase replicates the DNA downstream of the primers. By using a heat resistant polymerase from an Archaea species that lives at high temperatures, it is possible to rapidly cycle this iterative heating/cooling process, doubling the amount of desired segment of DNA each time. This technology makes possible the exponential amplification of entire DNA molecules or any specific region of DNA for which bracketing primers can be generated [49].

For visualization of the PCR product, a sample of the DNA mixture is loaded on a gel. A charged molecule will be accelerated when it is placed in an electric field. Positively charged molecules will move towards negative electrodes and vice versa. By placing the mixture of molecules of interest in a medium and subjecting them to an electric charge, the molecules will migrate through the medium and separate from each other. How fast the molecules will move both depends on their charge and their size, because bigger molecules experience more resistance from the medium. The resulting procedure, called *electrophoresis*, involves putting a spot of the mixture to be analyzed at the top of a polyacrylamide or agarose *gel*, and applying an electric field for a period of time. Then the gel is stained so

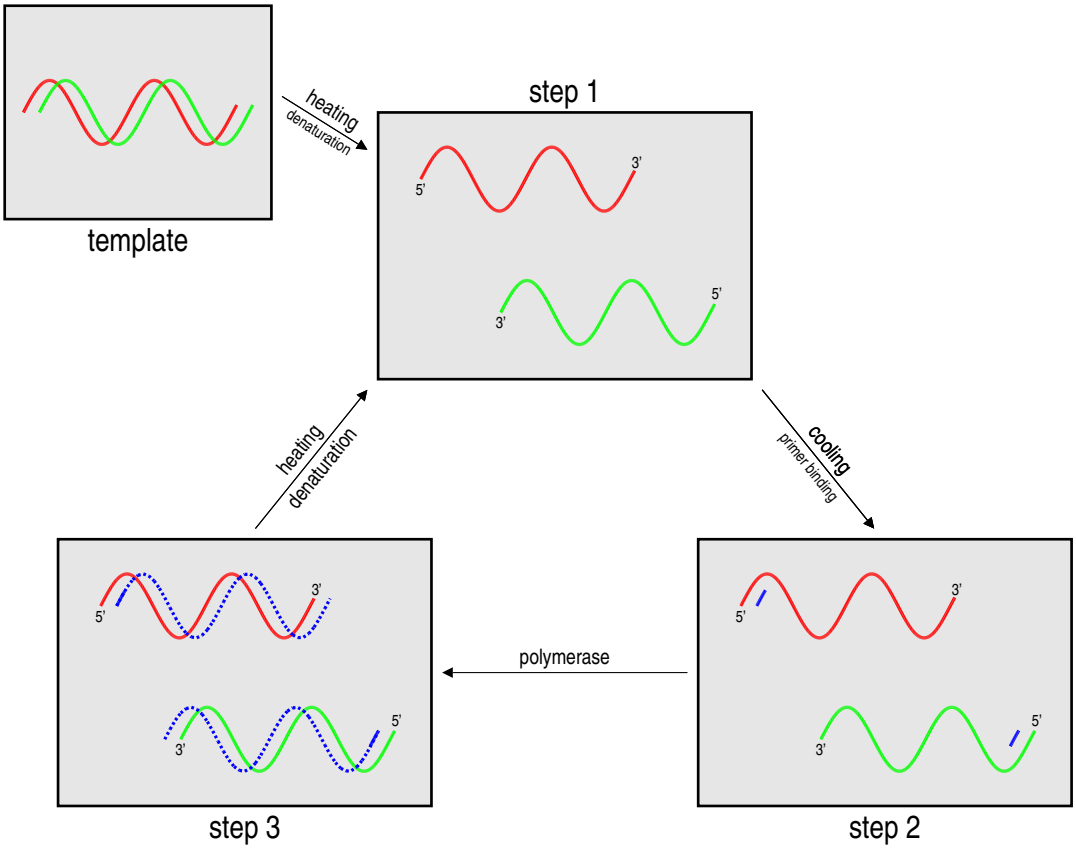


Figure 4.1: Replication of the template DNA via PCR. During step 1, the DNA double helix is denaturated so that each strand is accessible. After cooling the mixture in step 2, the primers bind to the loose DNA strands in order to allow subsequent binding of nucleotides. During step 3, the initial strands are copied by extending the primers, and the entire process can repeat all over again.

that the molecules become visible. These stains appear as stripes along the gel, and are called *bands*. The location of the bands on the gel are proportional to the charge and size of the molecules in the mixture. The intensity of the stain is an indication of the amount of the particular molecule in the mixture. If the molecules are all the same charge, or have charges proportional to their size (as, for example, DNA does) then electrophoresis separates them purely by size. Often, several mixtures are run simultaneously on a single gel. This allows for easy calibration to standards or visual comparison of the contents of different mixtures, showing for example the absence of a particular molecular component in one. The adjacent, parallel runs on the same gel are sometimes called *lanes*. The higher the similarity of the two lanes compared, the closer their cleavage pattern. For some example photographed electrophoresis gels we refer to Figures 4.2 and 4.3, where the remains of the slots at which the original mixtures were loaded can still be detected on top of the photographed RFLP gel. In the first example gel, molecular weight markers loaded in lanes 1, 2, 10 and 18 were used as the calibration standards for normalization of the lanes loaded on physically separate gels, whereas for the same purpose a reference pattern of the *Aeromonas hydrophila* subsp. *hydrophila* type strain LMG 2844^{T} was included in lanes 1, 6, 13, 23, 26, 32, 39 and 47 of the second example gel.

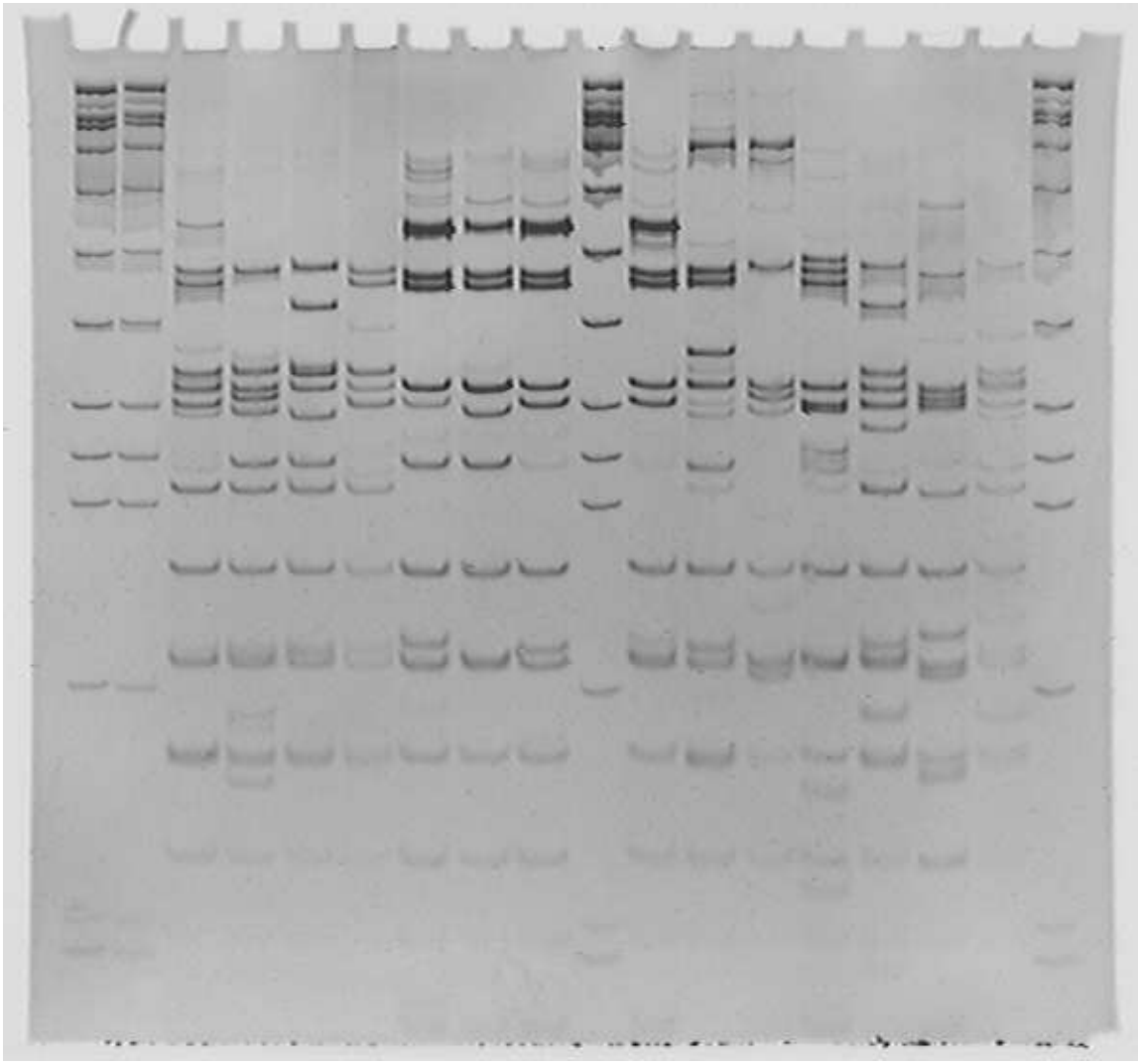


Figure 4.2: Photograph of a typical 16S RFLP gel (image kindly supplied by B. Lanoot). Lanes 1, 2, 10 and 18 contain a mixture of molecular weight markers, included for proper normalization of different gels.

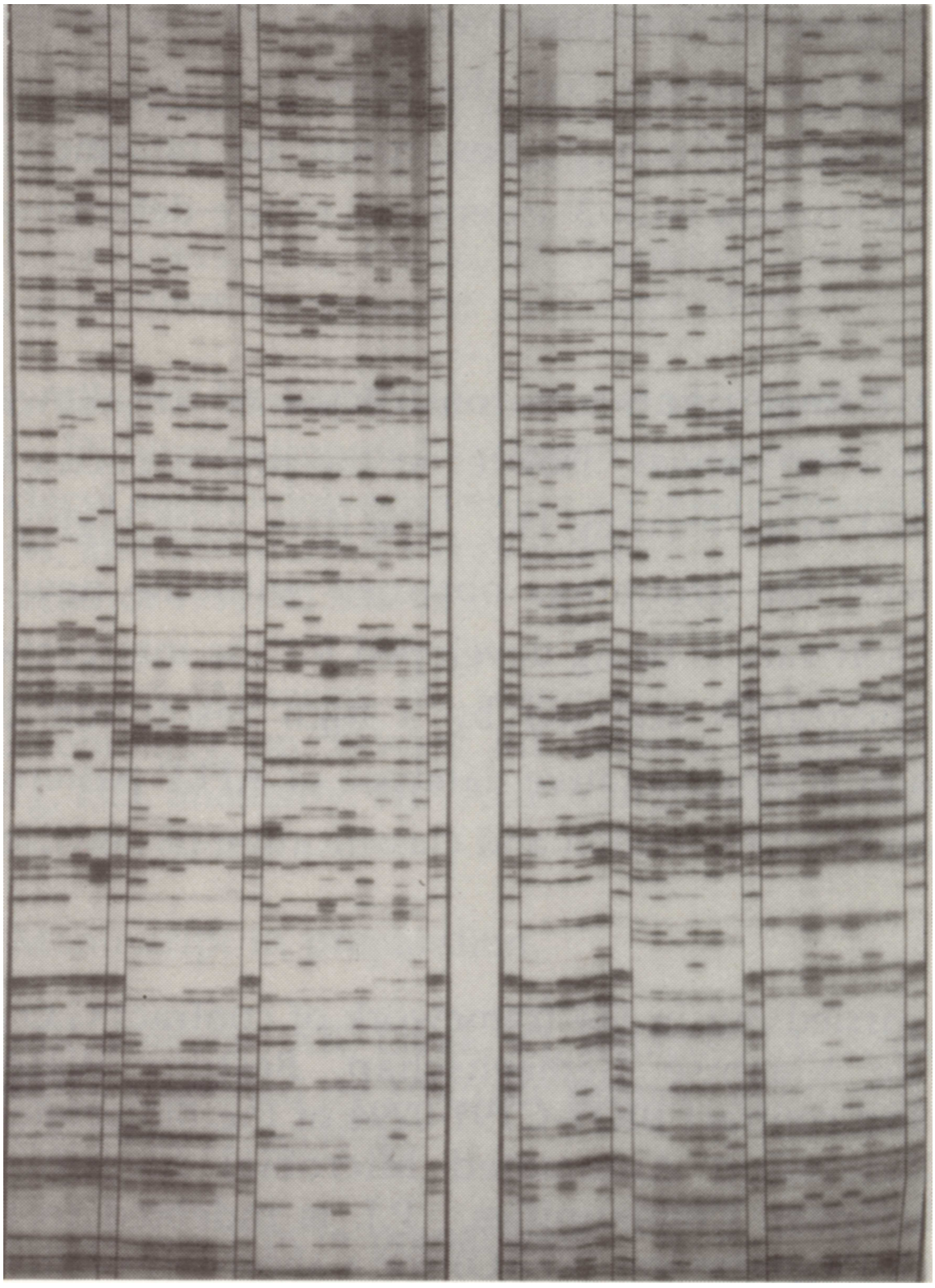


Figure 4.3: Scanned image of a radioactively labelled amplified fragment length polymorphism (AFLP) gel (image kindly supplied by G. Huys [50]). Lanes 1, 6, 13, 23, 26, 32, 39 and 47 contain a reference pattern of the *Aeromonas hydrophila* subsp. *hydrophila* type strain LMG 2844^{T}, included for proper normalization of the lanes loaded on physically separate gels.

4.2.1 AFLP

The DNA fingerprinting technique, known as amplified fragment length polymorphism (AFLP, [54]), is based on the selective amplification of genomic restriction fragments by polymerase chain reaction to differentiate bacterial strains at the subgeneric level, and consists of *i*) the digestion of total cellular DNA with two restriction enzymes and ligation of restriction halfsite-specific adaptors to all restriction fragments, *ii*) the selective amplification of these fragments with two PCR primers that have corresponding adaptor- and restriction-site sequences as their target sites and *iii*) the electrophoretic separation of the PCR products on a polyacrylamide or agarose gel. Only a subset of fragments will be amplified because the primers contain at their 3'-end one or more bases (the so-called selective bases) which are complementary to nucleotides flanking the restrictions sites and the reaction conditions are such that only perfectly matched primers will initiate DNA synthesis. In the original paper of Janssen *et al.* [54], radioactive labelling of one of the primers was suggested for visualization of the restriction fragments. A major improvement was obtained by switching from radioactive labelling to fluorescently labelled primers for detection of fragments in an automatic sequence apparatus [59], in which case fAFLP is used as acronym for the technique. Compared to other molecular techniques, probably the single greatest advantage of the AFLP technology is its sensitivity to polymorphism detection at the total-genome level, providing useful information about the short- and long-term evolution of bacterial strains. In order to fully understand the strengths and limitations of the AFLP technique and to get an appreciation of the kind of efforts required to produce this data used by computer scientists, a detailed description on the application of this molecular tool is given in subsection 4.11.2.

4.3 Comparison of fingerprint patterns

After electrophoresis, each organism is characterized by a banding pattern. Accordingly, these patterns are a direct reflection of the genetic relationships between the bacterial strains examined and, therefore, can be considered as molecular signatures allowing numerical analysis for characterization, classification and identification purposes. Although regarded somewhat deprecated since the wide-spread availability of inexpensive personal computers and special-purpose software packages for computational analysis, visual comparison is still a frequently used method for the interpretation of these electrophoresis patterns. Anyhow, it is always advisable to compare the results of computer analyses with the original gel electrophoresis patterns or their photographs in most (if not all) of the cases, even if computer-assisted analysis is thought to be the most objective method for interpretation [79]. This will probably avoid clear misinterpretations of the profiles due to unacceptable data reductions or ill-applied transformations during the numerical processing.

In order to enable digital processing of the molecular signature profiles, the genotyping fingerprint patterns are usually recorded by a densitometer or line scanner which measures the optical density at regular sample points along the gel tracks, or by a digital camera

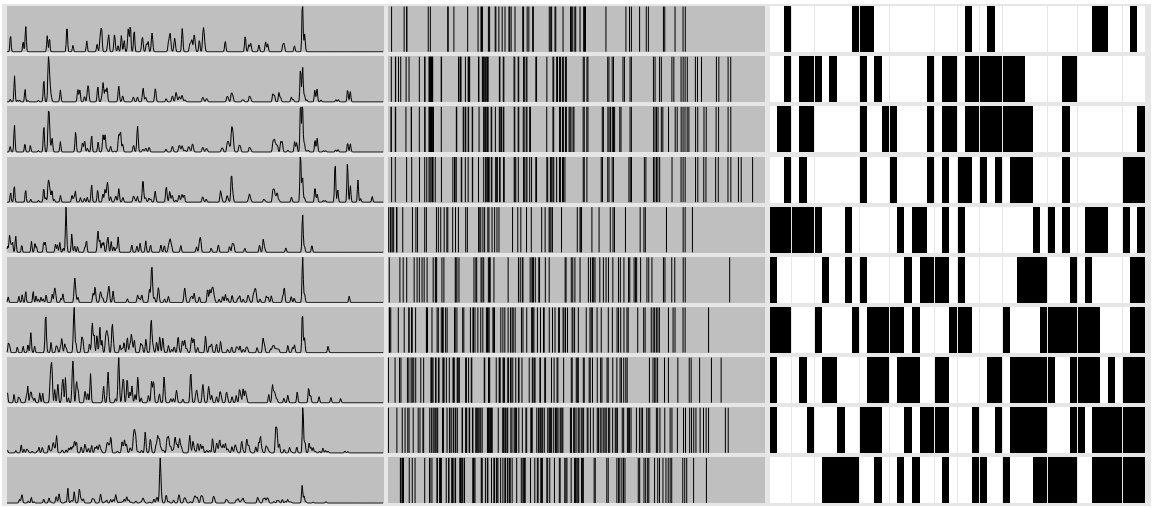


Figure 4.4: Examples of normalized genotyping fingerprint patterns represented as densitometric curves (left), band patterns (middle) and binary vectors (right).

that rasters the optical density of the whole gel surface. Once captured on a computer, there exist several transformation procedures for preprocessing the banding patterns into a workable numerical format. In essence, three digitized banding pattern representations are suitable for computational analysis: normalized densitometric curve representation, normalized band pattern representation and normalized (binary) vector representation (Figure 4.4). In order to achieve any of these representations, successive computational data transformations are required, as is schematically indicated in Figure 4.5. These include graphical enhancements of image quality and removal of artefacts of the electrophoresis process. After all, high background levels can heavily reduce the discriminatory strength of the fingerprinting techniques, hence several algorithms have been developed to subtract the background noise at different stages of the processing chain based on power series polynomial trend analysis [28], linear regression [53], Fourier smoothing of concave kernels [62] and rolling disk removal [101].

Normalization involves standardizing the length of the fingerprinting profiles to compensate for inevitable tiny fluctuations within and between gels, as the electrophoresis results are subject to experimental error. The latter are caused by small but notable variations in factors such as preparation of bacterial samples, chemical composition of the electrophoresis gel medium, exposure duration and strength of the electrical field during electrophoresis and deviations during gel digitization. To enable the normalization procedure, a reference bacterial extract or a mixture of molecular weight markers (containing purified proteins) is included at regular positions within the gel (external reference patterns), or loaded inline into each lane of the gel and revealed with a different color dye or hybridization probe (internal reference patterns). Digitized traces are aligned (brought to equal length) by three-point quadratic interpolation techniques [28]. This is achieved by aligning a number of stable and easily recognizable peaks on the references traces, whereby the intervals between these peaks are stretched or shrinked following the interpolation procedure. The non-reference traces are recalculated in the same way as the closest neighbouring reference trace, or as a gradual interpolation of their surrounding reference traces [115].

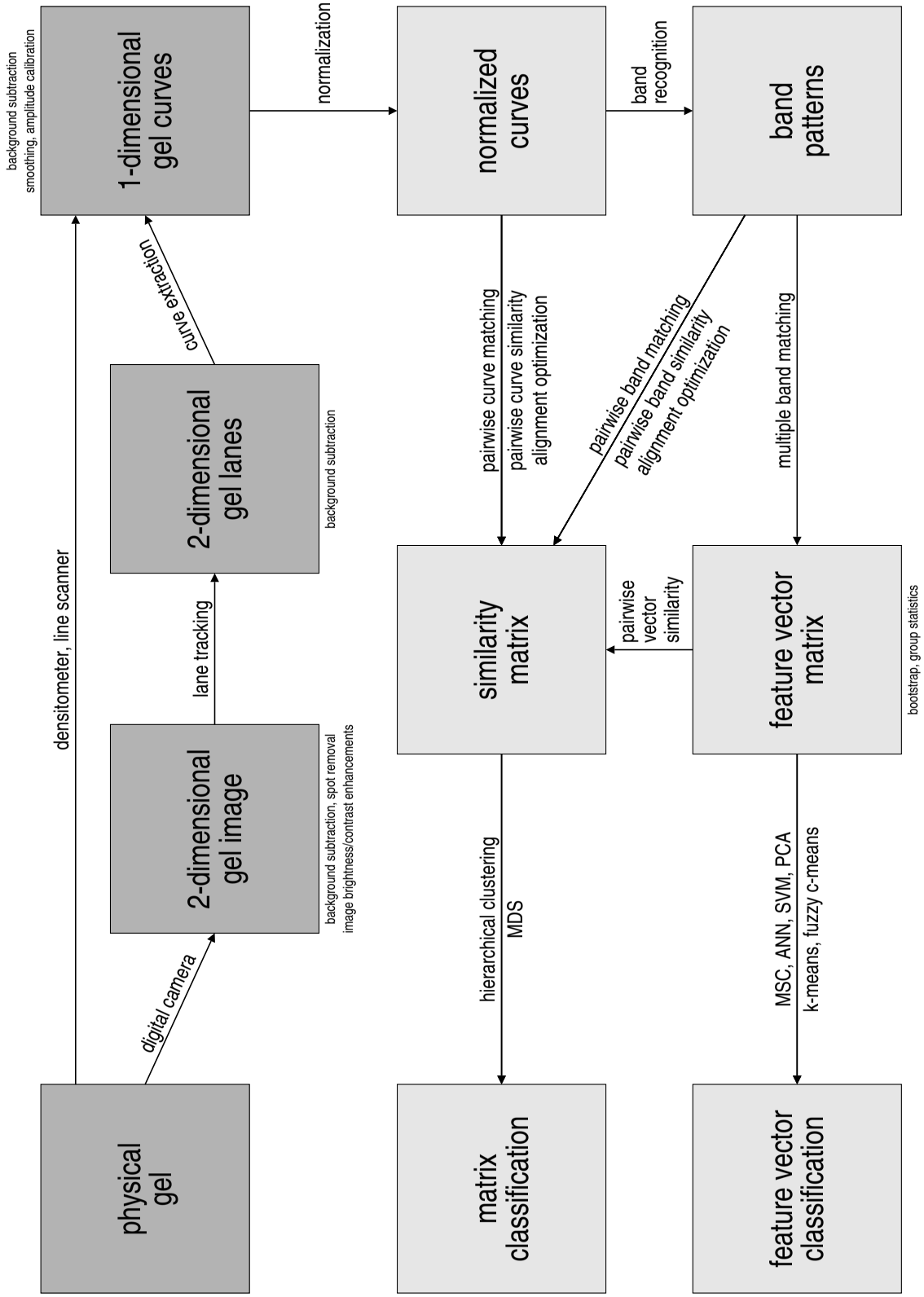


Figure 4.5: Successive processing steps during the analysis of phenotypic or genotypic fingerprinting patterns. MDS, multidimensional scaling; MSC, minimization of stochastic complexity; ANN, artificial neural networks; SVM, support vector machines; PCA, principal component analysis.

Once normalized one-dimensional densitometric curves are extracted for all gel lanes, an estimation of the similarity of two bacterial samples can be measured directly from their electrophoregrams. This pairwise curve matching is primarily applicable for the computational analysis of very complex banding patterns, such as those generated by polyacrylamide gel electrophoresis (PAGE) of cellular protein fingerprints, for which no simple band extraction can be performed without significant loss of the information content stored within the original profiles [79]. For these cases, the resulting fingerprint patterns are compared without any attempt to characterize individual electrophoretic bands. Several pairwise curve matching coefficients are discussed in greater detail in section 4.4.

Genotypic electrophoresis patterns frequently contain clearly defined peaks. In these cases, it is more suitable to further reduce the data into a set of peak positions and the corresponding integrated peak areas. The analysis of bands extracted from the genotyping fingerprint profiles is usually based on a formal *similarity model* imposed on a given set of fingerprint patterns [65, 97]. Such similarity models *i*) describe the relatedness of bands between different fingerprint patterns and *ii*) provide an agglomerative method to quantify this relatedness. Although large numbers of existing similarity and dissimilarity coefficients are potentially useful for quantifying the resemblance of genotyping fingerprint data, few alternatives are available for describing the relationship between bands of different fingerprint patterns; a process commonly referred to as *band matching* [81, 89, 115]. Given a set of fingerprint patterns, where each pattern is represented as a vector of band positions, two major band matching strategies can be distinguished.

A first possible band matching strategy is to define the necessary conditions for two bands of different fingerprints to be matched. As this kind of matching relations are in general non-transitive (meaning that if a band of pattern A matches with a band of pattern B, which in turn matches with a band of pattern C, the band of pattern A does not necessarily match with that of pattern C), these types of methods will be referred to as *pairwise* band matching algorithms. Three alternative pairwise matching methods – called simple pairwise band matching, closest pairwise band matching and first pairwise band matching – are further discussed in more detail in section 4.6.

Another family of band matching procedures consists of transforming the band position representation for each fingerprint pattern into a binary vector representation, so that band positions are mapped to vector indices. Each vector index thus corresponds to a class of common bands among the patterns. As the mapping procedure is the same for each individual fingerprint pattern, all patterns will be transformed to vectors of equal length. This latter type of method will thus be referred to as *multiple* band matching algorithms or *discretization* algorithms. A straightforward method for performing this transformation, called equal–width discretization, partitions the range of the profiles into n subintervals of equal width [78]. The BioNumerics software package (Applied Maths, Sint-Martens-Latem, Belgium) offers a more sophisticated data–dependent method for generating a vectorized representation of fingerprint patterns. Both methods are discussed in more detail in section 4.7. Finally, section 4.8 is completely dedicated to another member of this band matching family, which is an extension of the equal–width discretization with partially overlapping band classes. We have termed this method sliding window discretization.

Further enhancements to improve the comparability of gel lanes can be obtained by a repetitive lateral shift procedure carried out during the similarity calculation between individual pairs of traces. The technique iteratively searches for an optimal similarity fit along the x-axis of the fingerprinting profiles, by probing small additive [79, 115] and/or multiplicative [89] transformations of the profiles. However, these local alignment optimization methods are only applicable for pairwise pattern comparisons (both curve or band matching), and not for multiple band matching methods.

As a final remark, we want to note that in the context of DNA sequence analysis, the same terminology of *pairwise* aligned sequences [29, 34, 73, 96, 112] and *multiple* aligned sequences [3, 18, 35, 68, 120] is used to discriminate between non-transitive and transitive data representations used for matching corresponding bases in different pairs of DNA sequences. The multiple band matching process for fingerprint patterns, that transforms a non-transitive band representation into a transitive vector representation, can then be seen as the analogon of multiple sequence alignment that performs the same transformation for DNA sequence data.

4.4 Pairwise curve matching

For the computational analysis of two densitometric curves patterns $A(x)$ and $B(x)$, the curves are represented as a series of n sample points of the optical density along the x-axis

$$A^T = (a_1, a_2, \ldots, a_n) \quad \text{and} \quad B^T = (b_1, b_2, \ldots, b_n)\,. \tag{4.1}$$

The ith component of each vector is the score of the ith sample point as measured by the densitometer or extracted from the digitized gel image. The superscript T indicates the transposition of vectors. All vectors in this section are regarded as column vectors, while row vectors are represented as transposed column vectors. Two measures make sense for estimating the similarity between these two sampled curve vectors: the *cosine measure* and *Pearson's product moment correlation*. Each of them is scrutinized in the following subsections.

4.4.1 Cosine measure

In linear algebra, the *inner product* (or scalar product) of two vectors is given by

$$< A, B > = \; A^T B \; = \sum_{i=1}^{n} a_i b_i\,. \tag{4.2}$$

In statistics, this quantity is also known as the sum of cross products between A and B. The inner product of a vector with itself, $A^T A$, is known as the sum of squares for A. The square root of the sum of squares is the Euclidean norm or length of the vector, and is

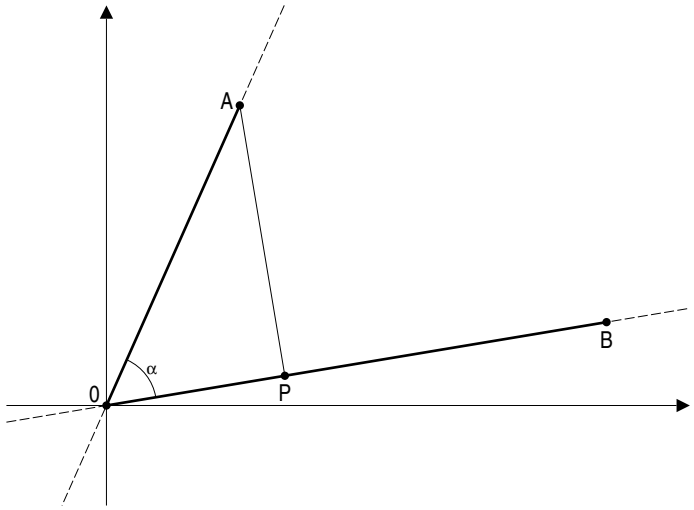


Figure 4.6: Illustration of the inner product and geometric interpretation of the cosine invariance.

conventionally written as $|A|$ or $||A||$. With this notation, an alternative expression for the inner product between A and B is

$$A^T B = |A||B| \cos\alpha \,, \tag{4.3}$$

where α is the angle between A and B in the n-dimensional space. In a two-dimensional space, this relation can be depicted as in Figure 4.6. The distance from O (the origin) to P is $|A| \cos\alpha$ as is well known from elementary geometry. This quantity is also the length of the orthogonal projection of A onto B. The inner product then is seen to be the product of the length of B and the length of the projection of A onto B. Solving equation (4.3) for the cosine of the angle α gives

$$\cos\alpha = \frac{A^T B}{|A||B|} = \frac{\sum_{i=1}^{n} a_i b_i}{\left(\left[\sum_{i=1}^{n} a_i^2\right]\left[\sum_{i=1}^{n} b_i^2\right]\right)^{1/2}} \,. \tag{4.4}$$

The cosine of the angle α can be regarded as a measure of the similarity between A and B, and its value is within the interval $[-1, 1]$. The cosine of the angle α for two collapsing vectors is 1 (or -1 if they have opposite directions), while the cosine for two orthogonal vectors is 0. Therefore, it is standard procedure to calculate the cosine similarity coefficient as COSINE$(A, B) = |\cos\alpha|$, which results in a similarity value within the unit interval. The more nearly parallel the two vectors, the greater is the cosine. The cosine similarity coefficient is independent of the length of the vectors, as is obvious from the geometry of Figure 4.6. Algebraically, let w_a and w_b be two non-zero scalar constants and define

$\hat{A} = w_a\, A$ and $\hat{B} = w_b\, B$. Then

$$\frac{\hat{A}^T \hat{B}}{|\hat{A}||\hat{B}|} = \frac{\sum_{i=1}^{n} (w_a\, a_i)(w_b\, b_i)}{\left(\left[\sum_{i=1}^{n} w_a^2\, a_i^2 \right] \left[\sum_{i=1}^{n} w_b^2\, b_i^2 \right] \right)^{1/2}} \tag{4.5}$$

$$= \frac{w_a\, w_b \sum_{i=1}^{n} a_i b_i}{|w_a\, w_b| \left(\left[\sum_{i=1}^{n} a_i^2 \right] \left[\sum_{i=1}^{n} b_i^2 \right] \right)^{1/2}} \tag{4.6}$$

$$= \text{sgn}(w_a\, w_b) \frac{A^T B}{|A||B|}\,, \tag{4.7}$$

where sgn$(w_a\, w_b)$ is the sign (+ or -) of the product of w_a and w_b. Given the absolute value in the expression of the cosine similarity measure, we have in a more compact notation that

$$\text{COSINE}(A, B) = \text{COSINE}(w_a\, A, w_b\, B)\,. \tag{4.8}$$

Thus, the cosine similarity coefficient is invariant to uniform multiplicative scaling. The geometric interpretation of this relation is also shown in Figure 4.6. Every point on the line OA and its projection in both directions (except for the origin O itself) is equivalent to A under the cosine measure, and likewise for the line OB and B. Thus, the cosine similarity coefficient is a many-to-one transformation which effectively ignores the relative magnitudes between the vectors, hence also the relative magnitudes between the densitometric curves that are compared.

4.4.2 Pearson's product moment correlation

The mean values for the feature vectors A and B representing the sampled densitometric curves are computed as

$$\bar{a} = \frac{1}{n} \sum_{i=1}^{n} a_i \quad \text{and} \quad \bar{b} = \frac{1}{n} \sum_{i=1}^{n} b_i\,. \tag{4.9}$$

Now, if the vector means are subtracted from the original scores, the vectors of centered scores are obtained as

$$\hat{A}^T = (a_1 - \bar{a}, a_2 - \bar{a}, \ldots, a_n - \bar{a}) \quad \text{and} \quad \hat{B}^T = (b_1 - \bar{b}, b_2 - \bar{b}, \ldots, b_n - \bar{b})\,. \tag{4.10}$$

Note that $\hat{A}$ and $\hat{B}$ have zero means. The inner product of the two centered vectors is called the *scatter* of A and B. The inner product of $\hat{A}$ with itself is the scatter of A or the sum

of squared deviations around $\bar{a}$. If the scatter is divided by the number of sample points n, then the covariance and variance are recognized as

$$\text{cov}(A,B) = \frac{\hat{A}^T\hat{B}}{n} = \frac{1}{n}\sum_{i=1}^{n}(a_i-\bar{a})(b_i-\bar{b})\,, \tag{4.11}$$

$$\text{var}(A) = \frac{\hat{A}^T\hat{A}}{n} = \frac{1}{n}\sum_{i=1}^{n}(a_i-\bar{a})^2\,. \tag{4.12}$$

The covariance of A and B is also known as the product moment of A and B. Likewise, var(A) is the product moment of A. Pearson's product moment correlation [75] between the vectors A and B is then defined as

$$r = r(A,B) = \left|\frac{\text{cov}(A,B)}{[\text{var}(A)\text{var}(B)]^{1/2}}\right| = \left|\frac{\sum_{i=1}^{n}(a_i-\bar{a})(b_i-\bar{b})}{\left(\left[\sum_{i=1}^{n}(a_i-\bar{a})^2\right]\left[\sum_{i=1}^{n}(b_i-\bar{b})^2\right]\right)^{1/2}}\right|\,. \tag{4.13}$$

Note again that absolute value is taken to limit the possible values of this measure to the unit interval. By comparing the equations (4.4) and (4.13) it is apparent that r is equal to the absolute value of the cosine of the angle between the centered vectors $\hat{A}$ and $\hat{B}$.

An alternative view of the Pearson product moment correlation can be established in terms of standardized vectors, in which the components are transformed as

$$a_i^* = \frac{(a_i-\bar{a})}{[(\text{var}(A)]^{1/2}}\,. \tag{4.14}$$

The vector A^* with components a_i^* has zero mean and unit variance. The inner product of A^* and B^* is then

$$A^{*T}B^* = \frac{\sum_{i=1}^{n}(a_i-\bar{a})(b_i-\bar{b})}{[\text{var}(A)\text{var}(B)]^{1/2}}\,, \tag{4.15}$$

so that $r = \frac{A^{*T}B^*}{n}$. The Pearson correlation is actually the absolute value of the covariance of the standardized vectors.

Since the product moment correlation is the cosine of the angle between the centered vectors, it inherits the multiplicative invariance of the cosine. In addition, it is also invariant under the uniform addition of a constant to each element of A or B. Let $a_i^+ = a_i + v$ for all i. Then the mean of the transformed variable is $\bar{a}^+ = \bar{a} + v$. Therefore,

$$a_i^+ - \bar{a}^+ = (a_i+v) - (\bar{a}+v) = a_i - \bar{a}\,. \tag{4.16}$$

Thus, the added constant is subtracted out in the process of centering the scores. The combined effect of these two forms of invariance is that Pearson's product moment correlation coefficient is invariant to any linear transformation, that is

$$r(A,B) = r(w_a\,A + v_a, w_b\,B + v_b)\,. \tag{4.17}$$

The Pearson correlation has a stronger form of invariance than the cosine because it is unaffected by the uniform addition of a constant to each element of a score vector. But this same property means that the correlation coefficient is less discriminating than the cosine as for given A and B there are many more members in the equivalence class of all linear transformations of A and B than in the equivalence class of all multiples of A and B.

The essential difference between the two measures is that the cosine is based on the original scores (deviations from the origin) while the correlation coefficient is based on centered scores (deviations about the mean). If the origin is well established and meaningfull, then the original scores have meaning in an absolute sense and the cosine is an appropriate measure of association. If the origin is arbitary or chosen for convenience, then the original scores are meaningfull relative to each other and to their mean, but not relative to the origin. In this case the correlation coefficient is an appropriate measure of association. Note that for densitometric curve patterns that represent molecular fingerprint profiles, background noise of the gel medium and other influencing factors might result in a linear transformation of the original signal. As the linear transformation can be different for lanes taken from different gels (or even from the same gel), the Pearson product moment correlation is the most frequently used similarity coefficient for calculating measures of association between fingerprint profiles represented as sampled curves.

4.5 Band matching

Let us now tackle the problem of fingerprint band matching in a more formal context. Given a set of m genotyping fingerprint patterns $\{B_i\}_{i=0}^{m-1}$, where each pattern is represented as a list of band positions, the total number of bands of pattern B_i is denoted by n_i. The kth band of pattern B_i is indicated as b_{ik} and the measured or derived position of this band is denoted by x_{ik}. Without loss of generality we can assume that the range of band positions is scaled to the unit interval, such that $0 \leq x_{ik} \leq 1$ for all $i \in \{0, \ldots, m-1\}$ and all $k \in \{0, \ldots, n_i - 1\}$, and that for all patterns the bands are ordered in increasing order of band position, such that for each band B_i it holds that $x_{ik} < x_{il}$ if $k < l$.

In spite of the numerous normalisation preprocessing steps applied to genotyping fingerprint data before the actual comparative analysis is executed [56, 115], small shifts in band positions cannot be avoided [89]. Therefore, all band matching methods discussed in this chapter have in common that they make use of the same kind of error tolerance parameter $\varepsilon \in [0, 1]$. The parameter ε is systematically referred to as the *position tolerance* of the band matching method, because it compensates for small drifts in the run length of the molecules during electrophoresis. In fact, the position tolerance is the maximal allowed shift between two bands to still consider them as being homologous, i.e. equivalent for comparison. The position tolerance ε parameter should be carefully selected for a given data set, as too large values lead to overfitting (many non-homologous bands in the same band class) and too small values to underfitting (many homologous bands in different band classes). Visual inspection of the band matching result on a series of duplicate band patterns generated for the same microbial sample may hereby avoid unexpected behaviour of

the computational results.

In the forthcoming sections, several alternative methods for performing band matching between genotypic fingerprint patterns are formally presented. As a means to illustrate the procedure for each of these methods, we will elaborate on a small example data set, containing only three genotyping fingerprint patterns with the following band position representations

$$\begin{aligned} B_0 &= [0.092, 0.167, 0.228, 0.236, 0.351, 0.424, 0.653, 0.787, 0.849, 0.921] \\ B_1 &= [0.096, 0.147, 0.237, 0.242, 0.355, 0.420, 0.427, 0.644, 0.655, 0.662, 0.783, 0.854] \\ B_2 &= [0.104, 0.244, 0.489, 0.562, 0.833, 0.856]\,. \end{aligned} \tag{4.18}$$

These example banding patterns are graphically represented in Figure 4.7.

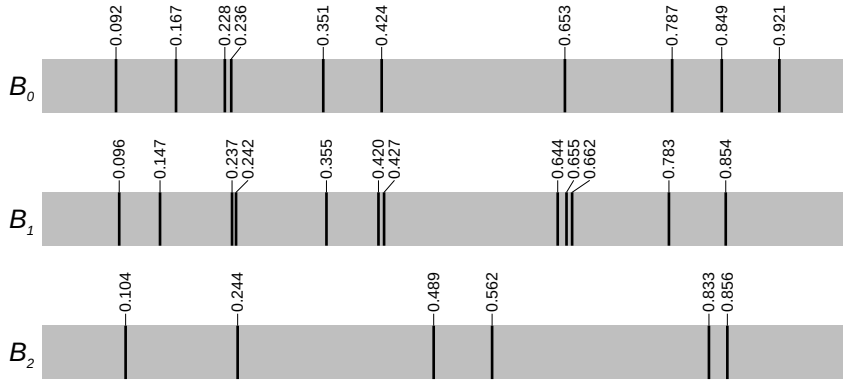


Figure 4.7: Graphical representation of the example band patterns given in (4.18).

4.6 Pairwise band matching

As is already stated in the name itself, pairwise band matching methods impose a similarity model onto a given set of banding patterns, by scoring the level of relatedness for each couple of patterns in the data set. This results in the construction of a similarity or dissimilarity matrix, as is shown in Figure 4.8. In the previous chapter we have presented several techniques to impose a hierarchical ordering on top of these matrices. Other methods such as multidimensional scaling (MDS, [15, 21]) or self-organizing maps (SOM, [60]) also can generate classifications for a given data set from the information in its calculated similarity or dissimilarity matrix. The actual agglomeration of two discrete band representations into an estimation of their overall relatedness is performed in two successive steps. First the homologous band are identified (i.e. bands that are regarded to represent equivalent molecules). This step is termed as band matching. Secondly, a similarity coefficient is applied to quantify the number of matched or common bands into an estimation of the global relatedness of the two patterns. Note that the scored homology in general is

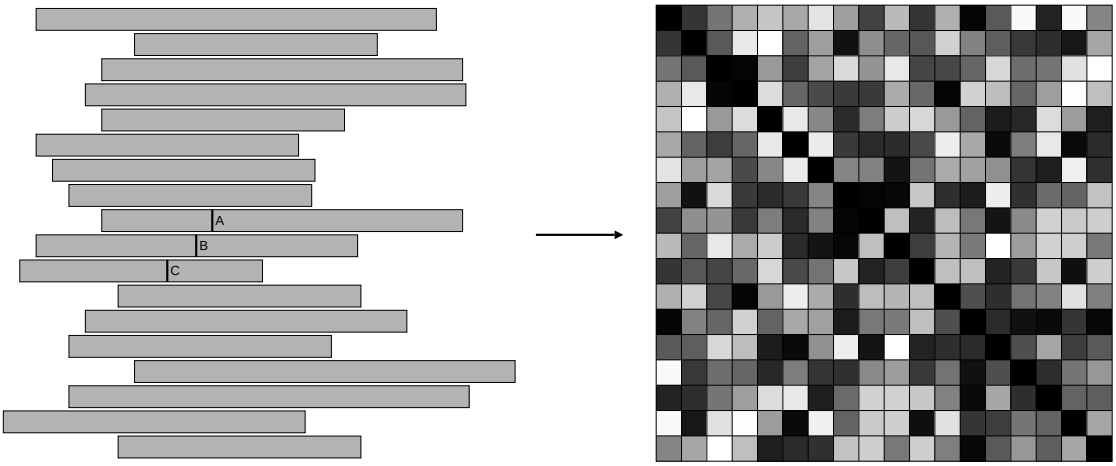


Figure 4.8: Pairwise band matching.

a non-transitive property of the bands, meaning that if a band of pattern A matches with a band of pattern B, which in turn matches with a band of pattern C, the band of pattern A does not necessarily match with that of pattern C. As a result, even if the applied similarity coefficient is known to produce similarity matrices that possess a weak form of transitivity (e.g. Łukasiewicz transitivity) for a given set of feature vectors [24], similarity matrices produced by pairwise band matching methods do not conform to the same property in all cases, as the compared vector representation of a given band pattern (if it exists) may vary over its different pairwise comparisons. Three alternative pairwise band matching methods are reviewed in this section: simple pairwise band matching, closest pairwise band matching and first pairwise band matching. For a survey on similarity coefficients we refer to [2, 97].

4.6.1 Simple band matching

Although regarded as the most basic amongst all pairwise band matching methods (hence the selection for its name), the *simple pairwise band matching method* has to our knowledge never been described in literature or applied for comparison of molecular banding profiles. But, as stated before, descriptions of the used band matching strategy in the scientific literature or in the documentation of special-purpose software packages are generally pretty vague or completely missing. In the simple pairwise band matching method, each band of pattern B_i will match with each band of pattern B_j within a distance defined by the position tolerance ε. Formally, we define the simple band matching function $m_s(b_{ik}, b_{jl})$ by

$$m_s(b_{ik}, b_{jl}) = \begin{cases} 1 & \Leftrightarrow \quad |x_{ik} - x_{jl}| \leq \varepsilon \\ 0 & \Leftrightarrow \quad |x_{ik} - x_{jl}| > \varepsilon \,. \end{cases} \tag{4.19}$$

Accordingly, we say that band b_{ik} is matched with band b_{jl} only if $m_s(b_{ik}, b_{jl}) = 1$. The function $m_s(b_{ik}, B_j)$ denotes the number of bands of pattern B_j that are matched with band

b_{ik}, and is defined by

$$m_s(b_{ik}, B_j) = \#\{b_{jl} | l = 0, \ldots, (n_j - 1) \wedge m_s(b_{ik}, b_{jl}) = 1\}\,. \tag{4.20}$$

Following this definition, we say that the band b_{ik} of pattern B_i is matched with a band of pattern B_j only if $m_s(b_{ik}, B_j) \geq 1$. The number of bands of pattern B_i that are matched with bands of pattern B_j is given by the value of the function $m_s(B_i, B_j)$, defined as

$$m_s(B_i, B_j) = \#\{b_{ik} | k = 0, \ldots, (n_i - 1) \wedge m_s(b_{ik}, B_j) \geq 1\}\,. \tag{4.21}$$

It should be clear that although the definition of m_s given in (4.19) is symmetric (i.e., $m_s(b_{ik}, b_{jl}) = m_s(b_{jl}, b_{ik})$), the same property does not hold in general for counting the number of matched bands of the patterns B_i and B_j as defined in (4.21). This means that the number of matched bands when comparing two lanes does not have to be the same for both lanes. We will discuss the relationship of this pairwise method with the newly introduced sliding window discretization method further on in this chapter.

Figure 4.9 shows a graphical illustration of the simple pairwise band matching method applied to all pairs of patterns from the example given in (4.18) with position tolerance $\varepsilon = 0.01$, where all couples of bands having $m_s(b_{0k}, b_{1l}) = 1$ are connected by means of a dotted line. Note that although there is a match between the first bands of the patterns B_0 and B_1, and between the first bands of the patterns B_1 and B_2, there is no match between the first bands of B_0 and B_2 for the chosen setting of the position tolerance ε. Table 4.1 shows the band matching table for the simple pairwise band matching method, containing the values $m_s(B_i, B_j)$ for the fingerprint patterns of the example. This band matching matrix indeed lacks symmetry.

4.6.2 Closest band matching

In the *closest pairwise band matching* method each band of pattern B_i will match with another band of pattern B_j within a distance defined by the position tolerance ε, only when these bands are *mutually closest* to each other. By mutually closest bands b_{ik} and b_{jl} we mean that there is no band in lane B_j closer to band b_{ik} of lane B_i than band b_{jl}, and vice versa. Formally, we define the closest band matching function $m_c(b_{ik}, b_{jl})$ by

$$m_c(b_{ik}, b_{jl}) = \begin{cases} 1 \quad \Leftrightarrow \begin{cases} |x_{ik} - x_{jl}| \leq \varepsilon \\ (\forall k' \in \{0, \ldots, n_i - 1\} \setminus \{k\})(|x_{ik} - x_{jl}| < |x_{ik'} - x_{jl}|) \\ (\forall l' \in \{0, \ldots, n_j - 1\} \setminus \{l\})(|x_{ik} - x_{jl}| < |x_{ik} - x_{jl'}|) \end{cases} \\ 0 \quad \text{otherwise}\,. \end{cases} \tag{4.22}$$

Accordingly, we say that the bands b_{ik} and b_{jl} are matching bands of patterns B_i and B_j only if $m_c(b_{ik}, b_{jl}) = 1$. The function $m_c(b_{ik}, B_j)$, defined by

$$m_c(b_{ik}, B_j) = \#\{b_{jl} | l = 0, \ldots, (n_j - 1) \wedge m_c(b_{ik}, b_{jl}) = 1\}\,, \tag{4.23}$$

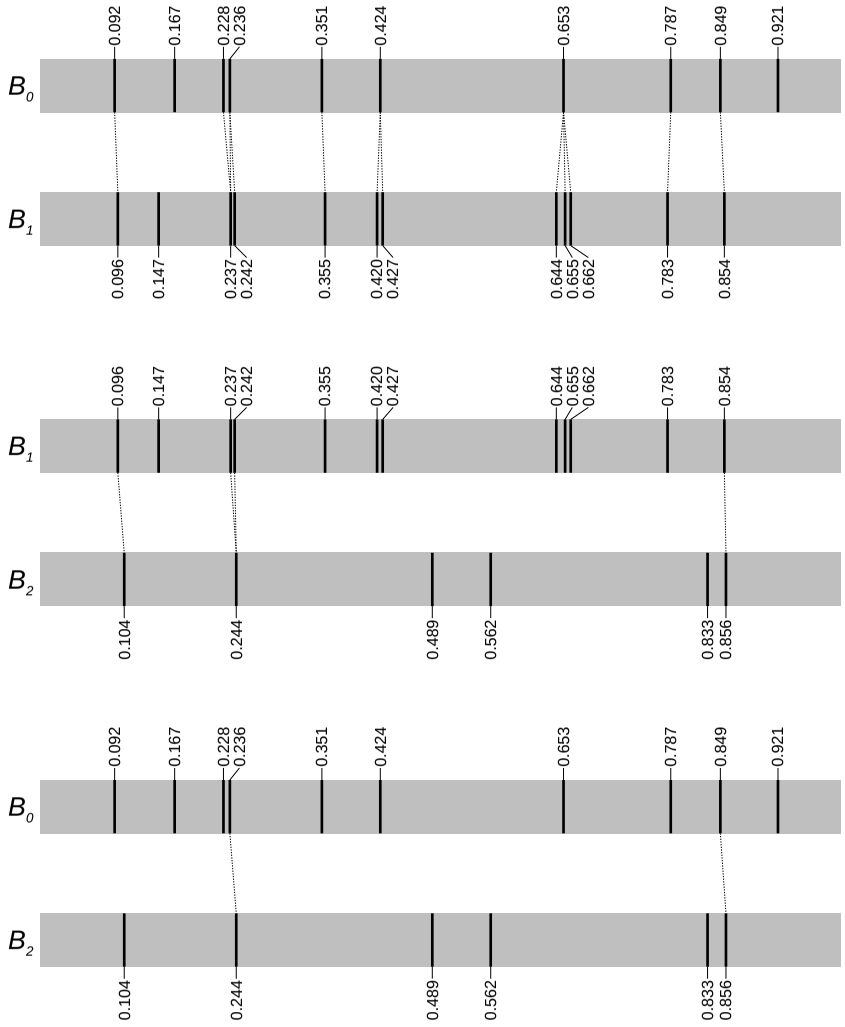


Figure 4.9: Simple pairwise band matching results for all pairs of patterns from example (4.18), with position tolerance $\varepsilon = 0.01$.

	B_0	B_1	B_2
B_0	10	8	2
B_1	11	12	4
B_2	2	3	6

Table 4.1: Matching table showing $m_s(B_i, B_j)$ for the banding patterns of example (4.18), with position tolerance $\varepsilon = 0.01$.

denotes the number of bands of pattern B_j that are matched with the band b_{ik} of pattern B_i. Following this definition, we say that the band b_{ik} of pattern B_i is matched with a band of pattern B_j only if $m_c(b_{ik}, B_j) \geq 1$. It is easy to verify that $m_c(b_{ik}, B_j) \in \{0, 1\}$, which means that a band will maximally match with one other band. Together with the symmetric property of $m_c(b_{ik}, b_{jl})$ this justifies the terminology that b_{ik} and b_{jl} are *common* bands of the patterns B_i and B_j, often used in literature where this band matching method is applied.

Although the conditions for matching bands are more strict in this method compared with those of the simple band matching method, the additional advantage of using the closest band matching method is that there is a possible vector representation for comparing a pair of band patterns, so that all vector matching coefficients can be applied for quantifying the pattern relatedness. We can define the function $m_c(B_i, B_j)$, denoting the number of bands the patterns B_i and B_j have in common, by

$$m_c(B_i, B_j) = \#\{b_{ik} | k = 0, \ldots, (n_i - 1) \wedge m_c(b_{ik}, B_j) = 1\} . \tag{4.24}$$

Note that the functions m_c given in (4.22) and (4.24) are now both symmetric. The closest pairwise band matching method was first introduced by Salamon et al. [89], as the band matching component of their align-and-count matching method for automated processing of restriction fragment length polymorphism (RFLP) fingerprint patterns of *Mycobacterium tuberculosis* strains. The authors also concluded from their experience with analyzing extensive sets of RFLP banding patterns, that the errors in fragment length measurements are proportional to the fragment length itself, i.e. larger drifting of the molecules towards the end of the gel lanes. As a result, the value of the position tolerance ε can be made variable along the range of the banding patterns, which is applicable for any pairwise band matching method.

Figure 4.10 shows a graphical representation of the closest pairwise band matching method applied to all pairs of patterns from example (4.18) with position tolerance $\varepsilon = 0.01$, where all couples of bands having $m_c(b_{0k}, b_{1l}) = 1$ are connected by means of a dotted line. Table 4.2 shows the band matching table for the closest pairwise band matching method, containing the values $m_c(B_i, B_j)$ for the fingerprint patterns of the example. This band matching matrix is now indeed symmetric.

4.6.3 First band matching

In the *first pairwise band matching* method, the bands of patterns B_i and B_j are scanned in increasing band position order. The first pair of bands that is within a distance defined by the position tolerance ε will match, and the focus is shifted to the next two bands. If no match is found between the two current bands in the scanning process, then the focus is shifted to the next of the leftmost of the two bands under investigation. This procedure is repeated until all bands of one of the given profiles have been completely scanned. More formal, band b_{ik} is matched with band b_{jl} only if $m_f(b_{ik}, b_{jl}) = 1$, where the pseudo-code

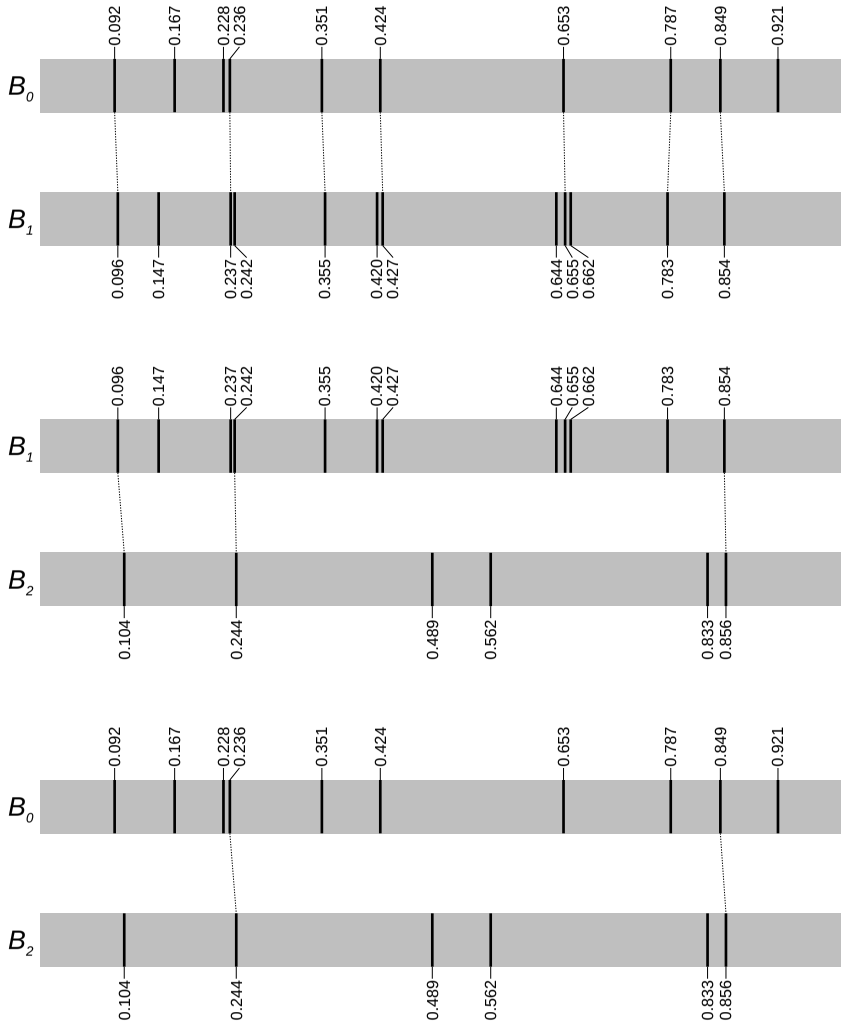


Figure 4.10: Closest pairwise band matching results for all pairs of patterns from example (4.18), with position tolerance $\varepsilon = 0.01$.

	B_0	B_1	B_2
B_0	10	7	2
B_1	7	12	3
B_2	2	3	6

Table 4.2: Matching table showing $m_c(B_i, B_j)$ for the banding patterns of example (4.18), with position tolerance $\varepsilon = 0.01$.

of the algorithm for defining the first pairwise matching function $m_f(b_{ik}, b_{jl})$ is given by

```
for k from 0 to n_i-1 do
   for l from 0 to n_j-1 do
      m_f(b_ik,b_jl) = 0;
k = 0; l = 0;
while (k < n_i and l < n_j) do
   if |x_ik - x_jl| ≤ ε then
      m_f(b_ik,b_jl) = 1;
      m_f(b_jl,b_ik) = 1;
      k = k + 1; l = l + 1;
   else
      if (x_ik < x_jl) then k = k + 1;
      else l = l + 1;
```

The matching functions $m_f(b_{ik}, B_j)$ and $m_f(B_i, B_j)$ are defined as in (4.23) and (4.24). These functions have the same transitivity and symmetry properties as their counterpart matching functions m_c of the closest pairwise band matching method, so that this method also has a possible vector representation for comparing each pair of band patterns. Further on in this chapter, we will find an indication that the BioNumerics software package (Applied Maths, Sint-Martens-Latem, Belgium) has implemented the first pairwise band matching method for the comparison of banding patterns, although this is not explicitly mentioned in the supplied user documentation.

Figure 4.11 shows a graphical representation of the first pairwise band matching method applied to all pairs of patterns from example (4.18) with position tolerance $\varepsilon = 0.01$, where all couples of bands having $m_f(b_{0k}, b_{1l}) = 1$ are connected by means of a dotted line. Note the difference in behaviour of the first and closest pairwise band matching methods for matching the common bands of patterns B_0 and B_1 in the range [0.228,0.242]. Where the closest pairwise band matching method matches the middle couple of the four bands, the first band matching method matches both leftmost and rightmost bands. Which of the two interpretations is more correct may depend on the nature of the banding patterns under investigation. Table 4.3 shows the band matching table for the first pairwise band matching method, containing the values $m_f(B_i, B_j)$ for the fingerprint patterns of the example. This band matching matrix indeed also shows a symmetric property.

4.7 Multiple band matching

Multiple band matching is an alternative technique to impose a similarity model upon a given set of fingerprinting profiles. Each multiple band matching technique describes the band relatedness for genotyping fingerprint patterns by means of a transformation of the band position representation B_i for each pattern into a (binary) vector representation V_i, where all vectors are of equal length n, as is depicted in Figure 4.12. In this representation

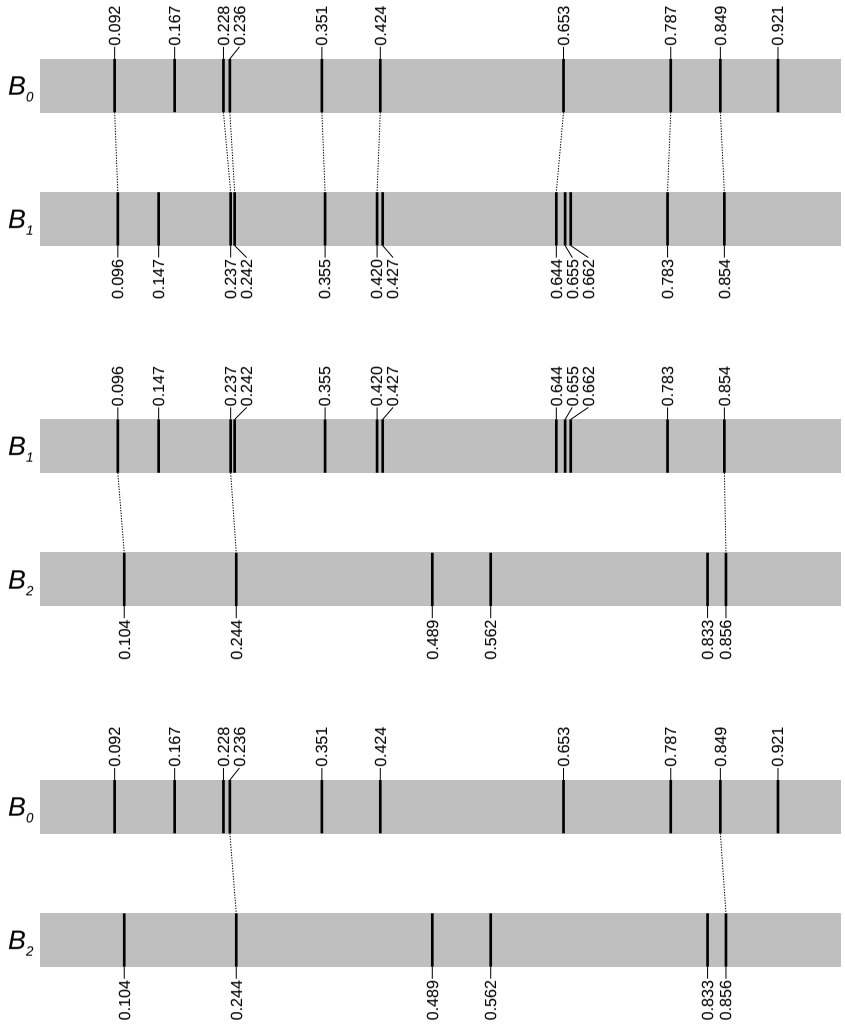


Figure 4.11: First pairwise band matching results for all pairs of patterns from example (4.18), with position tolerance $\varepsilon = 0.01$.

	B_0	B_1	B_2
B_0	10	8	2
B_1	8	12	3
B_2	2	3	6

Table 4.3: Matching table showing $m_f(B_i, B_j)$ for the banding patterns of example (4.18), with position tolerance $\varepsilon = 0.01$.

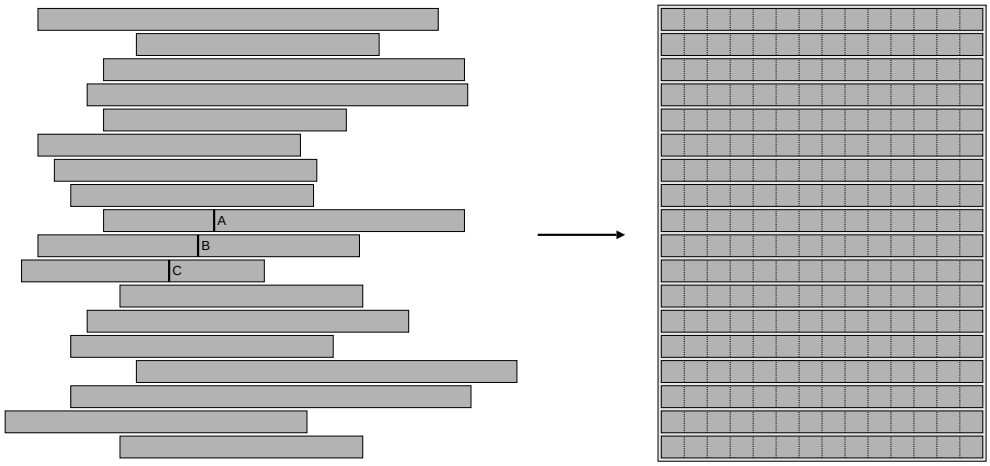


Figure 4.12: Multiple feature mapping: in the context of nucleotide sequences the mapping of homologous base pairs is called multiple sequence alignment, while in the context of molecular fingerprint patterns, we speak of normalization when aligning the band profiles, and band matching when mapping the individual bands.

the vector indices are treated independently, but values at corresponding vector positions are comparable between the different fingerprint patterns. Note that although the final representation will be generally the same for all multiple band matching methods, the meaning of the corresponding indices of vector patterns with respect to the original band patterns will be different for each method. Which multiple band matching methods are applicable and which choice is the best at hand will both depend on the original band patterns (losslessness of transformed information) and the numerical methods one wants to apply upon the binary vector representations (meaning of vector indices).

Although a sheer endless series of similarity coefficients exists for the transformation of the feature vector representation of a set of patterns into a similarity matrix, such that all classification methods that are applied in combination with pairwise band matching methods are equally applicable for analyzing the multiple band matching results, a number of other classification and validation tools work directly upon the vector representation. Bootstrap analysis measures the robustness of classification methods against small perturbations of the orginal data set, by application of a *sampling with replacement* procedure. This artificial resampling process can only be mimicked on vector representations. But apparently the most interesting opportunity offered by a vector representation of the original dataset, is the possiblity to calculate and compare several group statistics for a selection of patterns: *i*) group representatives can be determined as average feature vectors, *ii*) the homogeneity or heterogeneity of a group can be measured as the overall variance of the group, and *iii*) features that most abundantly differentiate a given group from other groups can be traced for, amongst many others. Several well-known classification and identification methods explicitely make use of these group statistics for the execution of their procedure: minimization of stochastic complexity, k-means, fuzzy c-means, principal component analysis (PCA), artificial neural networks (ANN), support vector machines (SVM). In this chapter we review three multiple band matching techniques, being equal-width discretization,

histogram-based discretization and a new method called sliding window discretization.

4.7.1 Equal–width band matching

By far being the simplest and most frequently used discretization method, the *equal–width multiple band matching* method partitions the unit interval $\Delta = [0, 1]$ into n non-overlapping contiguous subintervals $\Delta_{k,(k=0,\ldots,n-1)}$ of equal width ε, defined by

$$\begin{cases} \Delta_k & = & [\Delta_k^l, \Delta_k^r[= [k\varepsilon, (k+1)\varepsilon[, \quad k = 0, \ldots, n-2 \\ \Delta_{n-1} & = & [\Delta_{n-1}^l, \Delta_{n-1}^r] = [(n-1)\varepsilon, n\varepsilon] \,. \end{cases} \tag{4.25}$$

As was the case for all previous band matching methods presented in the current chapter, the interval width ε will be called the position tolerance of the equal-width method and it can be derived from the vector length n as

$$\varepsilon = \frac{1}{n} \,, \tag{4.26}$$

or vice versa. Each band of fingerprint pattern B_i will be mapped to the unique interval Δ_k that contains this band, so that the vector V_i can be constructed as

$$y_{ik} = \begin{cases} 1 & \text{if } (\exists j \in \{0, \ldots, b_i - 1\})(x_{ij} \in \Delta_k) \\ 0 & \text{otherwise} \,. \end{cases} \tag{4.27}$$

This results in a many-to-one mapping between bands in the band pattern representation and vector indices in the vector representation. Therefore, we say that the band:index relationship for this method is a N:1 relationship. A similar discretization method, known as *equal-frequency discretization*, divides the unit interval into n subintervals such that the total number of features of all patterns is equally distributed over the different bins. Because this method seems less applicable for the vector transformation of fingerprint patterns, we will not go deeper into this technique here. A serious drawback of the equal-width discretization method for the vectorization of molecular banding patterns is that homologous bands that are located relatively close together in light of the position tolerance ε, might still end up in neighbouring band classes, given the sharp boundaries drawn between these non-overlapping classes. Nevertheless, this method has sporadicly been applied in studies involving molecular fingerprint patterns where vector representations were needed [78], which has definitely not contributed to the popularity of multiple band matching for fingerprint patterns.

Application of the equal–width discretization method to the fingerprint patterns of the example given in (4.18), with the vector length set to $n = 14$ (hence $\varepsilon = 0.071$), results in the vector representation shown in Figure 4.13, where each vector index k is labeled with its corresponding subinterval Δ_k. It is standard procedure to discard the vector indexes that are zero for all generated patterns, either directly from the vector representation or during similarity and group statistics calculations.

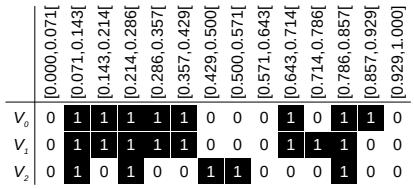


Figure 4.13: Equal-width discretization of the fingerprint patterns from example (4.18) into $n = 14$ band classes.

4.7.2 Histogram–based band matching

The *histogram–based multiple band matching* method given here is implemented in the BioNumerics software package (Applied Maths, Sint-Martens-Latem, Belgium; personal communication with P. Vauterin), where it is simply termed *band matching*. This method uses a data-dependent one-to-one mapping between bands in the band pattern representation and vector indices in the vector representation. This means that each band is mapped to one and only one vector index and maximally one band is mapped to a certain vector index for each fingerprint pattern. Therefore, we say that the band:index relationship is a 1:1 relationship. The method uses the following algorithm for mapping band positions to vector indices.

1. Create a band–appearance histogram for all band positions of all fingerprint patterns B_i, where each band contributes to each histogram bin within the position tolerance region (ε-region) of the band location. In the BioNumerics implementation, the contribution to the histogram is constant over the whole ε-region, but other schemes are equally possible, such as triangular contributions where the contribution is one at the band position and zero at a distance from the band position greater than or equal to the position tolerance ε.

2. If the optimization parameter is used, each fingerprint profile is shifted within the optimization interval such that the sum of the histogram values at the band positions of the profile reaches a maximum.

3. Create a new band class at the position where the histogram reaches its maximal value. This band class will correspond with a vector index in the vector representation of the fingerprint patterns.

4. For each fingerprint pattern, the closest unassigned band is assigned to the newly created band class if there is such a band of the pattern within the given position tolerance ε.

5. Recreate the band–appearance histogram in the same way as described in step 1, only taking into account the currently unassigned bands.

6. While there are unassigned bands, repeat the procedure from step 3 onwards.

The histogram–based multiple band matching method results in very compact vector representations, having no indices that are zero–valued for all patterns. On the other hand, it

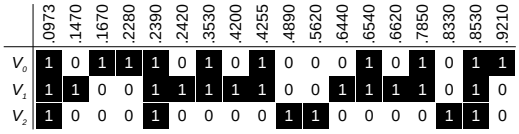


Figure 4.14: Histogram–based multiple discretization of the example fingerprint patterns, with the parameter ε set to 0.01 and no optimization.

heavily depends on the idea that each band position in the band representation of the fingerprint corresponds perfectly with a band in the original pattern. For very complex fingerprint patterns however, it is not always obvious to extract clearcut band positions from the original fingerprint pattern using an automated band extraction algorithm, the naked eye, or a combination of both techniques. In fact, for many fingerprinting techniques the notion of a band is rather artificial taken into account the different interpolation effects that occur during the electrophoresis. The histogram–based algorithm thus fails in situations where the resolution of bands within a pattern is very close to or even lower than the lateral shift due the normalisation and band extraction errors, because the algorithm forces two bands of the same fingerprint pattern that are within the position tolerance ε into different band classes. A highly similar algorithm is implemented within the LecPCR software package presented by Mougel *et al.* [72].

Applying the histogram–based multiple band matching algorithm to the example given in (4.18) with the position tolerance parameter ε set to 0.01 and no optimization, results in the vector representation of the fingerprint patterns as given in Figure 4.14. In this representation, each vector index k is labeled with the average band position of all bands associated to the corresponding band class. For convenience of comparison with the original banding patterns, the vector indices are reordered in increasing order of average band position.

4.8 Sliding window discretization

In the previous section we have indicated how application of the equal-width method for discretization of molecular fingerprint patterns suffers from drawing sharp boundaries around its non-overlapping band classes. Histogram–based multiple band matching, a discretization method that introduces the concept of overlapping band classes, on the other hand, is unsuccessful in dealing with complex banding patterns as it forces each band into a separate band class, hereby neglecting the putative artefacts of prior band extraction procedures that may distort peaks in the original scanned profile. After all, our experience with band extractions for several types of molecular fingerprinting techniques learns that peaks of nearly identical size or peaks showing shoulders in their densitometric curve are often inconsistenly extracted as one single band or as multiple closely located bands for different fingerprint patterns, both by human and computer-assisted band extraction algorithms. The human brain is surprinsingly capable of compensating for such discrepancies when performing visual comparisons, whereas computational methods often fail to have built-in capabilities to explicitely accomodate for this kind of inconsistencies. Therefore,

we introduce sliding window discretization as a new method for multiple band matching that combines the strengths of the two previous methods and can be made insensitive to most artefacts of the band extraction procedures by meticulous selection of its parameters.

As the sliding window discretization method is yet another member of the family of multiple band matching algorithms, it also transforms the band position representation B_i of each fingerprint pattern into a (binary) vector representation V_i, where all vectors are of equal length n. The binary value at the kth index of vector V_i will be noted as y_{ik}. The sliding window discretization method defines n subintervals $\Delta_{k,(k=0,\ldots,n-1)}$ of the unit interval $\Delta = [0, 1]$ as

$$\begin{cases} \Delta_k & = & [\Delta_k^l, \Delta_k^r[= [k\delta, k\delta + \varepsilon[, \quad k = 0, \ldots, n-2 \\ \Delta_{n-1} & = & [\Delta_{n-1}^l, \Delta_{n-1}^r] = [(n-1)\delta, (n-1)\delta + \varepsilon] \,. \end{cases} \tag{4.28}$$

These subintervals Δ_k are of equal width ε. The position tolerance parameter ε of the method must be chosen such that

$$\frac{1}{n} \leq \varepsilon < 1 \,, \tag{4.29}$$

and the parameter δ, called the *resolution* of the method, must be chosen in respect to the parameters n and ε such that

$$\delta = \frac{1-\varepsilon}{n-1} \,. \tag{4.30}$$

When $\delta = \varepsilon = \frac{1}{n}$, the sliding window discretization method is reduced to the equal–width method that is described in subsection 4.7.1. For all other values of the parameter ε, one has $\delta < \varepsilon$, which means that the subintervals Δ_k are overlapping. This discretization method thus scans the full range of the fingerprint patterns through a window of length ε at some regular intervals with intermediate distance δ. Each band of fingerprint pattern B_i will be mapped to all intervals Δ_k that contain this band, so that the vector V_i can be constructed as

$$y_{ik} = \begin{cases} 1 & \text{if } (\exists j \in \{0, \ldots, b_i - 1\})(x_{ij} \in \Delta_k) \\ 0 & \text{otherwise} \,. \end{cases} \tag{4.31}$$

This results in a many-to-many mapping between bands in the band pattern representation and vector indices in the vector representation. Therefore, we say that the band:index relationship for this method is a N:N relationship.

There is another interpretation for the sliding window discretization method, which relates it to the histogram–based multiple band matching method described in subsection 4.7.2. In this interpretation, the unit interval [0,1] is partitioned in equal–width subintervals of length δ, and the bands contribute to all subintervals that are within a distance defined by the position tolerance ε of the actual band position. As such, the bands are not seen as located on a fixed point but merely represent a region around the band position. This point of view also explains why we call δ the resolution of the sliding window discretization method. For some applications the introduction of dependencies among the features may be unwanted.

Application of the sliding window discretization method onto the fingerprint patterns of the example given in (4.18), with the vector length set to $n = 40$ and the position

	Jaccard(B_i, B_j)	Dice(B_i, B_j)	Jeffrey's x(B_i, B_j)
m_s	-	$\frac{m_s(B_i,B_j)+m_s(B_j,B_i)}{n_i+n_j}$	$\frac{1}{2}\left(\frac{m_s(B_i,B_j)}{n_i}+\frac{m_s(B_j,B_i)}{n_j}\right)$
m_c	$\frac{m_c(B_i,B_j)}{n_i+n_j-m_c(B_i,B_j)}$	$\frac{2m_c(B_i,B_j)}{n_i+n_j}$	$\frac{1}{2}\left(\frac{m_c(B_i,B_j)}{n_i}+\frac{m_c(B_i,B_j)}{n_j}\right)$
m_f	$\frac{m_f(B_i,B_j)}{n_i+n_j-m_f(B_i,B_j)}$	$\frac{2m_f(B_i,B_j)}{n_i+n_j}$	$\frac{1}{2}\left(\frac{m_f(B_i,B_j)}{n_i}+\frac{m_f(B_i,B_j)}{n_j}\right)$
m_{ij}	$\frac{m_{11}}{m_{11}+m_{10}+m_{01}}$	$\frac{2m_{11}}{2m_{11}+m_{10}+m_{01}}$	$\frac{1}{2}\left(\frac{m_{11}}{m_{11}+m_{10}}+\frac{m_{11}}{m_{11}+m_{01}}\right)$

Table 4.4: Difference in similarity coefficient implementations when combined with alternative band matching algorithms.

tolerance set to $\varepsilon = 0.05$, results in the vector representation as shown in Figure 4.15. The resolution of the sliding window discretization method δ is 0.0244 for this case. Again, the vector indexes that are zero for all generated patterns can be discarded either directly from the vector representation or during similarity and group statistics calculations, in order to control the total length of the vectors.

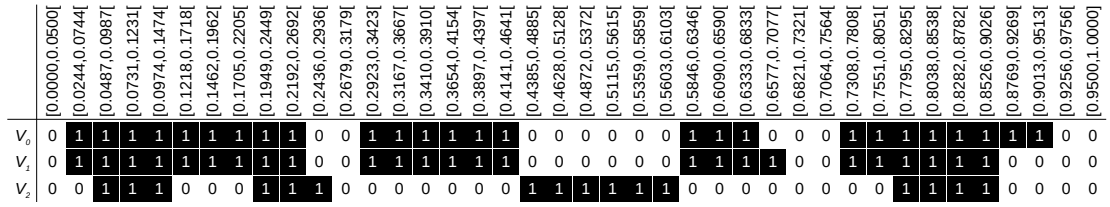


Figure 4.15: Binary vector representation resulting from application of the sliding window discretization method on the band patterns given in (4.18), with the vector length set to $n = 40$ and the position tolerance set to $\varepsilon = 0.05$.

4.9 Band pattern similarity quantification

The choice of a suitable band matching strategy might be strongly dependent on the type of the fingerprint patterns under investigation and the nature of the analysis one wants to perform. Moreover, the chosen band matching method also influences the applicability and implementation of the similarity coefficients for quantifying the strength of relatedness between pairs or groups of patterns. Table 4.4 gives a summary of the differences in the implementation of the Jaccard [52], Dice [25] and Jeffrey's x similarity coefficients for all band matching methods presented in this chapter. The first three rows correspond with simple, closest and first pairwise band matching respectively, whereas the shown similarity coefficients are identically implemented for all multiple band matching algorithms, as is indicated in the last row of the table.

For two binary feature vectors resulting from any of the multiple band matching algorithms, most similarity measures are based on frequency counts of the four combinations of corresponding index values: m_{11} represents the number of vector indices where both vectors score 1, whereas m_{00} is the number of vector indices that are 0 for the two vectors. The value m_{10} indicates how many vector indices are one in the first vector and zero in the second vector, while m_{01} is the dual case.

As the closest and first pairwise band matching methods associate a given band with at most one band of another pattern, the pairwise matching of any two banding patterns can be depicted as well using a vector representation. Because of this, the same implementation of the similarity coefficients can be used as for the case of multiple band matching methods, with the sole exception that the frequency value m_{00} is always zero. In Table 4.4, the corresponding similarity coefficient implementations are therefore presented in terms of the terminology introduced in subsections 4.6.2 and 4.6.3. Note however that the vector representations of a given band pattern might differ between different pairwise comparisons involving the pattern. As a result, the Jaccard and Dice similarity coefficients will no longer produce Łukasiewicz transitive similarity matrices, as they do for feature vector representations [24].

Because the first pairwise band matching function $m_s(B_i, B_j)$ gives up the symmetric property caused by the fact that one band can be matched with many other bands and vice versa, no vector representation can represent the simple pairwise band matching, and there is no clear translation of the frequency counts. The Dice and Jeffrey's x similarity coefficients, however, offer an opporunity to associate both values $m_s(B_i, B_j)$ and $m_s(B_j, B_i)$ within the same agglomerative measure, in order to maintain symmetry within the similarity matrices. An analog expression for the Jaccard coefficient does not exist.

4.10 Minimization of stochastic complexity

4.10.1 Stochastic complexity principles

According to Rissanen [86] the best model to explain a given set of data is the one which minimizes the sum of *i*) the length in bits of the description of the model, and *ii*) the length in bits of the description of the data within the model. This follows *Occam's Razor*, the principle that tells us not to introduce more concepts than necessary to explain observed facts. Classifying a collection of items according to some method (classification model) amounts to encoding information about the data. Applied to the classification problem, Rissanen's principle therefore yields that the best classification of a set of items is the one which requires the least number of bits to code the classification with respect to the model chosen and to code the data within the classification. The relevant mathematical quantity describing the code length is that of *stochastic complexity* (SC) [86]. The best classification is thus the one that minimizes SC.

Suppose that each item is represented by a binary vector $X = (x_1, \ldots, x_d)$ of length d. As classification model we choose a mixture

$$\lambda_1 p_1 + \cdots \lambda_k p_k \quad (\lambda_1 + \lambda_2 + \cdots + \lambda_k = 1, \lambda_i \geq 0) \tag{4.32}$$

of multivariate Bernoulli distributions

$$p_j(x) = \prod_{i=1}^{d} (1 - \theta_{ij})^{1-x_i} \theta_{ij}^{x_i} \tag{4.33}$$

Here p_j represents the jth class. The rationale for this choice is that among all probability distributions on the space B^d of binary d-vectors, compatible with the data in the sense that $\sum_{x \in B^d, x_{i=1}} p_j(x)$ is the relative frequency θ_{ij} of ones in the ith position in items in the jth class, the multivariate Bernoulli distribution (4.33) is the unique distribution that maximizes Shannon's entropy [39]. Gyllenberg *et al.* [40] showed that for this choice of classification model and a uniform prior distribution, the expression of stochastic complexity per strain is given by

$$\mathrm{SC} = \frac{1}{t} \left(\log \frac{k(k+1)\cdots(t+k-1)}{t_1! t_2! \cdots t_k!} + \sum_{j=1}^{k} \sum_{i=1}^{d} \log \frac{(t_j+1)!}{t_{ij}!(t_j - t_{ij})!} \right), \tag{4.34}$$

where k is the number of classes, t is the total number of items, t_j is the number of items in class j and t_{ij} is the number of items in class j with the ith component equal to 1. $\log$ denotes the logarithm to the base 2. The first term within the parentheses on the right hand side of (4.34) describes the complexity of the classification and the second term the complexity of the strains with respect to the classification. As such, stochastic complexity accomodates for overfitting by penalizing through the first term the possible classifications with many separate classes, and for underfitting by accounting for items that badly fit the class they are appointed to using the second term. A classification scheme that stores all items in a single class (extreme underfitting) has zero cost in the first term of the stochastic complexity, but maximal cost in the second term of the stochastic complexity. On the opposite side of the classification spectrum, grouping all items within their proper class (extreme overfitting) forces the stochastic complexity into maximal cost-assignment of the first term, but zero penalization in the second term. The optimal classification in terms of stochastic complexity often lies in between these two extremes. It should be noted that in the expression of stochastic complexity given in (4.34), the number of classes k is fixed. The optimal classification among all possible classifications with any number of classes can however be determined by using a global stochastic complexity minimization algorithm, such as the one implemented in the BinClass sofware package.

4.10.2 BinClass implementation

If there exists an objective function to measure the overall goodness of a given classification, a naive approach to find the optimal classification in terms of this objective function would simply calculate the cost function for each putative classification and choose the

one producing the optimum. However, the number of possibilities to group t items into k classes is given by Stirling's number of the second kind [1]

$$S_t^k = \frac{1}{k!}\sum_{j=0}^{k}(-1)^{k-j}\binom{k}{j}j^t\,, \tag{4.35}$$

which grows exponentially. For even the relatively small problem of grouping 25 items into 5 classes, the number of possibilities is the astounding quantity

$$S_{25}^5 = 2.436.684.974.110.751\,. \tag{4.36}$$

The problem is compounded by the fact that the number of groups is usually unknown, so that the number of possibilities is a sum of Stirling number. In the case of 25 items, we have a total number of different classifications given by

$$S_{25} = \sum_{k=1}^{25} S_{25}^k > 4\times 10^{18}\,. \tag{4.37}$$

Even small classification problems cannot be evaluated in this naive way even with the newest of powerful number-crunching computers, so more intelligent methods are needed to tackle the problem.

With stochastic complexity as evaluation function for classifications of binary vectors, the BinClass software package [42] scans the whole range or a user-selected slice of the possible cluster numbers $k \in \{1, 2, \ldots, t\}$, where t is the total number of items in the clustering problem. For a fixed number of classes k, a random classification is generated and iteratively improved by application of several variants of the *generalized Lloyd algorithm* (GLA, [33, 63]; also known as k-means or the Linde-Buzo-Gray algorithm) with the *Shannon code length* [20] as distance function. Because of the known problem of this method that it possibly gets stuck into local optima, a number of heuristics to avoid local optima are built into the software package, and additionally the procedure is repeated for a number of different start classifications for any fixed k. For all classifications produced by the k-means algorithm, the stochastic complexity is calculated, and the classification with the overal minimal value of the stochastic complexity is considered as the best classification by BinClass. This process is called *miminimization of stochastic complexity*. The Binclass software package also offers tools for minimization of SC using fuzzy c-means algorithms (*expectation maximization*), addition of new items to existing classifications (*cumulative classification*) and construction of hierarchical trees, based on the evaluation of stochastic complexity.

Minimization of stochastic complexity has been previously applied to the taxonomy of *Enterobacteriaceae* [36, 37, 38] and *Vibrionaceae* [43], based on some characteristic features of the phenotype observed from a representative sample of strains. Although the bits in the vectorized data sets used in these case studies might be dependent on each another, it is known that the previously described 'naive' Bayesian classification model performs quite well [36, 37, 38, 43]. The classification technique based on the minimization of stochastic complexity can be extended in such a way that dependencies between the bits

in the binary vector representation are taken into account [44]. However, this extension would dramatically add to the running time of the algorithm and is therefore not taken into considertation for the case study discussed in section 4.11.

4.10.3 A simple example

In this section we illustrate the basic idea of stochastic complexity minimization by a simple numerical example. Consider a collection of playing cards (not necessarily an ordinary pack), in which each card is characterized by $d = 4$ binary features: *i*) colour (red = 1, black = 0), *ii*) shape (heart-shaped, i.e., ♡ and ♠ = 1, non-heart shaped, i.e., ♢ and ♣ = 0), *iii*) parity (odd = 1, even = 0), and *iv*) royalty (K, Q and Kn = 1, other cards = 0). Thus, for instance, $9\heartsuit$ is represented by the feature vector $(1, 1, 1, 0)$.

Now assume that the collection to be classified by minimization of stochastic complexity consists of five cards, viz. $K\heartsuit$, $Kn\heartsuit$, $2\clubsuit$, $3\clubsuit$ and $4\clubsuit$. The four features chosen cannot distinguish between $K\heartsuit$ and $Kn\heartsuit$ or between $2\clubsuit$ and $4\clubsuit$, so essentially we have only three different cards out of $2^4 = 16$ possible. The reason why we consider such a small collection is that in real taxonomical applications the number of strains to be classified is only a tiny fraction of the totality of possible feature vectors.

The three feature vectors can be classified in only one way into $k = 1$ class, in three ways into $k = 2$ classes and in one way into $k = 3$ classes. For all possible classifications we have calculated the stochastic complexity using formula (4.34). The result is summarized in Table 4.5, from which we see that the classification separating the hearts from the clubs is the optimal one as evaluated by measuring the stochastic complexity. For more extensive data sets, it is of course unfeasible to enumerate all possible classifications and calculate the stochastic complexity of each classification. The implementation in the BinClass software package finds a reliable approximation of the least stochastic complexity value $\mathrm{SC}_{\min}(k)$ for any possible number of k classes. The overall SC-minimum is then obtained by choosing the classification into k classes resulting in the least of numbers $\mathrm{SC}_{\min}(k)$.

number of classes (k)	classification	stochastic complexity
1	c_1={$K\heartsuit$,$Kn\heartsuit$,$2\clubsuit$,$3\clubsuit$,$4\clubsuit$}	4.72551
2	c_1={$K\heartsuit$,$Kn\heartsuit$}, c_2={$2\clubsuit$,$3\clubsuit$,$4\clubsuit$}	**4.36634**
	c_1={$K\heartsuit$,$Kn\heartsuit$,$3\clubsuit$}, c_2={$2\clubsuit$,$4\clubsuit$}	5.00033
	c_1={$K\heartsuit$,$Kn\heartsuit$,$2\clubsuit$,$4\clubsuit$}, c_2={$3\clubsuit$}	5.70689
3	c_1={$K\heartsuit$,$Kn\heartsuit$}, c_2={$2\clubsuit$,$4\clubsuit$}, c_3={$3\clubsuit$}	5.19578

Table 4.5: Stochastic complexity for the different classifications of the example.

4.10.4 Finding the optimal α-cut for hierarchical classifications

A crucial issue when evaluating the dendrograms resulting from hierarchical cluster analysis, is the dilemma of selecting the number of clusters in the final solution [70], or, using the terminology of the previous chapter, finding the optimal α-cut for deriving an equivalence relation from the hierarchical grouping. In the research domain of microbial taxonomy, it is common practice to include a selection of type and reference strains into the set of strains under investigation, followed by visual delineation of the cluster cut off level based on the intuition of the researcher and his prior knowledge on the distribution of these reference strains, in order to attain clusters that correspond with the bacterial species concept. This bacterial species definition is usually described by empirical criteria. DNA-DNA cross-hybridization of $\geq$70% has been suggested to indicate that two bacteria belong to the same species [122]. Incited by the recommendations of the latest report of the *Ad Hoc Committee on Reconciliation of Approaches to Bacterial Systematics* [100], it has now been demonstrated by many authors that bacteria with cross-hybridization levels of $\geq$70% have a 16S rDNA sequence similarity of $\geq$97%. The whole discussion of the bacterial species concept in terms of fixed levels of similarity between strains stems from some experimental observations that demonstrate the existence of discontuities between groups of data [14]. However, the reports of the ad hoc commission lack a detailed description of the experimental methods, similarity measures and their parameters settings for quantifying the degree of relatedness used for conceptualizing the bacterial species. As a result, these sharp quantitive boundaries can only be used as rule of thumb, rather than as a strict definition.

In analogy to the observation of discrete natural boundaries between species, many procedures for automated delineation of the number of clusters in a specific data set have been proposed in the scientific literature. When a hierarchical clustering method is used, these procedures are often called *stopping rules* [69]. In a broader context, an objective function for evaluating the overall goodness of a classification is called an *index*. In general, one can discriminate two families of evaluation functions. A first class of indices estimates the classification of a set of items by means of the distance or similarity values calculated between any pair of items. Well-known members of this family are the C index [48], the gamma index [12], the G(+) index [87] and the point-biserial correlation [69]. The optimal number of clusters is found where the above measures attain their minimum or maximum value. A second group of indices is based on some group statisics directly calculated from the feature vectors. This family includes the Ball-Hall index [11], Calinski-Harabasz index [17], Davies-Bouldin index [22], gap statistic [111], Hartigan index [45], Krzanowski-Lai index [61], negative log-likelihood [121], Ratkowsky-Lance index [84] and silhouette index [55], among many others. A comprehensive survey of methods for estimating the number of clusters is given by Milligan [69, 70].

Stochastic complexity, defined in (4.34), can also be added to the latter family of methods for the special case of evaluating classifications of binary feature vectors. For the example outlined in subsection 4.10.3, calculation of the transitive closure (single linkage clustering) upon the similarity matrix of simple matching ($\approx$ Hamming distance) values leads to three possible α-cuts. Evaluation of the stochastic complexity for each of these cut-

ting levels would also point out the optimal classification for this example, using a smaller number of calculations of the objective function. More research would be needed to compare the behaviour of stochastic complexity for delineating 'natural' clusters, in respect to other evaluation functions.

4.11 Application to the taxonomy of *Vibrionaceae*

In this case study we evaluate the application of minimization of stochastic complexity in combination with sliding window discretization, as a viable strategy for the classification of bacterial genotypic fingerprinting patterns. To this means, we work on an fAFLP data set of 507 strains belonging to the family *Vibrionaceae*. Thompson *et al.* [103] present complete background information on this data set, together with a classification of the banding patterns based on Ward's hierarchical clustering algorithm [119]. The goal of this study is to test the robustness of the hierarchical approach by comparison with a classification derived using a method from a different mathematical family.

For generating a classification based on the minimization of stochastic complexity, the BinClass software package [42] is used. As this classification method is limited to the classification of fixed–length binary vectors, we initially need to transform the fAFLP molecular fingerprinting data into binary vector format. Several alternatives were evaluated for this purpose, in order to minimize the influence on the original similarity model used by Thompson *et al.* [103]. Sliding window discretization produced the most conservative discretization for the data set under investigation. The classification generated by the BinClass software package is compared in great detail with the original classification derived from hierarchical clustering, in order to find out where the classification methods agree and disagree.

4.11.1 Ecological and taxonomical traits of the family *Vibrionaceae*

Since the very first discovery and isolation of the species *Vibrio cholerae* (the causative agent of cholera) in the last half of the 19th century, a tremendous amount of research on the biology of vibrios (*i.e. Vibrionaceae* strains) has been done. The family *Vibrionaceae* has been under extensive investigation during the last few decades, making it by far the best documented marine taxon [58]. According to the most recent outline of Bergey's Manual of Systematic Bacteriology [13], the genus *Vibrio* (51 species), along with *Allomonas* (1 species), *Catenococcus* (1 species), *Enterovibrio* (1 species), *Grimontia* (1 species), *Listonella* (2 species), *Photobacterium* (7 species) and *Salinivibrio* (1 species), form the family *Vibrionaceae* which belongs to the γ-*Proteobacteria* [27, 99]. Members of this family are Gram negative, usually motile rods, mesophilic and chemoorganotrophic, facultative anaerobes and inhabitants of brackish, estuarine and pelagic waters and sediments. Vibrios are in high abundance in the marine environment and may participate in the nutrient cycling [26, 46, 82]. Moreover, they form the dominant culturable microbiota in and/or on marine

organisms, e.g., corals, fish, molluscs, seagrass, sponges, shrimps, and zooplankton, where these vibrios probably play an important role in digestion and nutrition [71, 85, 90]. Several *Vibrio* species are human and animal pathogens, while others form a serious threat to fish, shellfish and corals [8, 88]. The use of *Vibrio* strains as probiotics has been reported [5, 114], although in this respect the role of certain *Vibrio* species, e.g. *V. alginolyticus*, is still controversial.

With the advent of new molecular techniques, major taxonomic modifications have been proposed recently. Additional to *i*) the discovery of novel species [31, 32, 92, 93] and *ii*) the transfer of existing species to other new or existing genera [95, 113], *iii*) the most prominent flux in the taxonomical reorganisation is the subdivision of the original family into four separate families, i.e., *Salinivibrionaceae* (comprising the genus *Salinivibrio*), *Enterovibrionaceae* (comprising the genera *Enterovibrio* and *Grimontia*), *Photobacteriaceae* (comprising the genus *Photobacterium*) and *Vibrionaceae* (comprising all the *Vibrio* species except for the *V. fischeri* group). While the important (re)allocation of strains into species was achieved by molecular techniques such as AFLP and rep-PCR, the (re)organisation of the higher taxonomic ranks was primarily based on phylogenetic analysis of concatenated 16S rRNA, recA [102] and rpoA [110] gene sequences. So far, the whole-genome sequences of six vibrios (i.e. *V. cholerae* [47], *V. parahaemolyticus* [67], *V. vulnificus* [19, 57], *V. fischeri* [116] and *P. profundum* [117]) are available and at least another two (i.e. *V. lentus* and *V. salmonicida*) are under way. Complete genome sequences have shown that horizontal gene transfer, gene duplication and decay, and other rearrangements are probably driving forces in the evolution of genomes, which might shed a different light on bacterial taxonomy. An elaborated review on the biodiversity of vibrios and the latest taxonomical developments is given in [109].

4.11.2 fAFLP fingerprinting on selection of bacterial strains

Within the framework of this study we have analyzed a total set of 507 fluorescent amplified fragment length polymorphism (fAFLP, [54]) fingerprint patterns from isolates of the family *Vibrionaceae*, including 386 isolates originating from the marine aquacultural environment harvested between 1985 and 2001. This data set is identical with that studied by Thompson *et al.* [103], and all strains used in this study are listed in Appendix D. Strains were grown on Marine Agar 2216E (Difco Co., USA) at 27°C (*V. fischeri, V. logei, V. tapetis, V. salmonicida* and *V. wodanis* were grown at 18–19°C) for 24 hours, except for *V. cholerae* which was grown at Brain Heart Infusion Agar (Difco Co., USA). All strains included in this study are deposited in the BCCM^{TM}/LMG Bacteria Collection at the Ghent University. Approximately 0.01g of bacterial cells were harvested for DNA extraction, following the technique described previously by Pitcher *et al.* [77]. Concentration and purity of the DNAs were estimated measuring optical densities at 234, 260 and 280 nm using a Uvicom 941+ spectrophotometer (Kontron Instruments, Italy). DNA integrity was verified on a 1% Agarose gel in 1$\times$ TAE buffer (40mM Tris/Acetate, 1mM EDTA, pH 8.0).

Fluorescent amplified fragment length polymorphism (fAFLP) template preparation was

carried out essentially as described by Janssen *et al.* [54]. It has been shown that the application of this high resolution genomic fingerprinting technique might have a tremendous impact on the study of the diversity, taxonomy and phylogeny of several bacteria [123]. Moreover, experience with this technique has made clear that similarity measures based on the AFLP patterns reflect well the DNA-DNA hybridisation measures between bacteria. Due to its high discriminatory power, AFLP can thus be used as an identification tool. To accomplish this for the given data set, one μg of high-molecular-mass DNA was digested with *Taq*I (5'TCGA3') and *Hind*III (5'AAGCTT3') (Amersham Pharmacia Biotech, Sweden), followed by ligation of restriction half-site specific adapters to all restriction fragments with T4 ligase (Amersham Pharmacia Biotech, Sweden). Templates were precipitated in a solution containing 50% Isopropanol and 1.25 M NH_4OAc and dissolved in 100 μl T0.1E buffer (10 mM Tris-HCL, 0.1 mM EDTA, pH 8.0). Two subsequent PCR amplifications were applied. For the pre-selective PCR-amplification, 5 μl of template was mixes with 0.6 μl H00-ABI primer (5'GACTGCGTACCAGCTT3'; 1 μM), 0.6 μl T00-ABI primer (5'CGATGAGTCCTGACCGA3'; 5 μM), and 18.7 μl of Amplification Core Mix (Applied Biosystems, USA). The amplification reactions were performed in a GeneAmp PCR System 9600 thermocycler (Applied Biosystems, USA) using the following temperature program: 2 min at 72°C and 20 cycles of 20 sec at 94°C, 30 sec at 56°C and 2 min at 72°C. Pre-selective products were diluted in 130 μl T0.1E buffer (10 mM Tris-HCL, 0.1 mM EDTA, pH 8.0). In the selective PCR-amplification, 2.0 μl of the diluted solution was mixed with 0.7 μl H01-6FAM primer (5'GACTGCGTACCAGCTTA3'; 1M), 0.7 μl T03-ABI (5'CGATGAGTCCTGACCGAG'; 5μM), and 10 μl of Amplification Core Mix. The H01-6FAM primer is fluorescently labelled, and the selective bases at the 3'-end are underlined. The temperature profile of the selective amplification was as follows: *i*) denaturation for 2 min at 94°C, *ii*) 10 cycles of : denaturation for 20 sec at 94°C, annealing at decreasing stringency at 67-n°C for 30 sec (with n the cycle number), and extension at 72°C for 2 min, *iii*) 20 cycles of : denaturation for 20 sec at 94°C, annealing at 56°C for 30 sec, and extension at 72°C for 2 min, and *iv*) final extension at 60°C for 30 min.

Separation of the selective PCR products was generated on 36cm denaturing polyacrylamide gels (4.25% Acrylamide, 6 M Urea in 1$\times$ TBE/89 mM Tris + 89 mM Boric acid + 2mM EDTA, pH 8.3) on an ABI Prism 377 DNA sequencer (Applied Biosytems, USA). Before loading 1 μl of the samples on the gel, 1.5 μl of the selective product was mixed with a loading buffer (0.75 μl deionised Formamide, 0.25 μl Blue Dextran, 50 mM EDTA solution, 0.5 μl GeneScan-500 TAMRA size standard and 0.5 μl GeneScan-2500 TAMRA size standard) and heated at 95°C for 3 min. The mix was kept on a thermobloc (-20°C) while the gel was being loaded. The data were registered during electrophoresis run at 51°C by the ABI Prism^{TM} Data Collection Software (Applied Biosystems, USA) for 3.5 hours. Tracking and normalization of the lanes were performed by the GeneScan 3.1 software package (Applied Biosystems, USA). The total number of bands of the fAFLP profiles ranged from 46 to 164, with a global average of 107 bands and a standard deviation of 23. More detailed band statistics for each separate class resulting from the classification based on the minimization of stochastic complexity, are included in Table 4.8.

In their original paper, Thompson *et al.* [103] have imported the normalized band patterns, containing fragments of 50 to 536 base pairs, into the BioNumerics 2.0 software

package (Applied Maths, Sint–Martens–Latem, Belgium) for further numerical analysis. Classification of the fAFLP fingerprinting profiles was performed using the Dice similarity coefficient s_D [25] and Ward's hierarchical clustering method [119]. For relaxed fragment comparison, a band position tolerance value of 0.5% was allowed to compensate for misalignment of homologuous bands due to technical imperfections. The Ward/Dice hierarchical clustering of the fAFLP band patterns of the 507 strains studied, resulted in 69 clusters (labelled A1,A2,...,A69) and 4 singleton fingerprint patterns (labelled U1,U2,U3 and U4) at a cut-off level of 45% similarity. This level of cluster delineation was selected based on previous studies concerning *Acinetobacter* [54], *Aeromonas* [51], *Bradyrhizobium* [123] and *Vibrio* [6, 7, 76] and prior knowledge about the distribution of the type strains within the data set. With a few exceptions, each actually recognised species showed a characteristic genome pattern and fell into a separate cluster. Visual inspection revealed that some clusters harboured more diverse isolates than other clusters, which were composed of highly related patterns, almost identical to each other. Strains with indistinguishable genomes were isolated from the same source at the same date and place, suggesting the occurrence of one particular clone.

For reproducibility control, the fAFLP fingerprint patterns of 70 representative strains were generated twice, starting from new DNA isolation. The 70 pairs of band patterns were numerically analysed and the mean Dice similarity value found for reproduced fingerprint patterns was $91\pm3\%$. This similarity value is in accordance with previous studies using AFLP [54, 123]. Strains clustering at the reproducibility level or higher were indistinguishable by fAFLP. Duplicate fingerprint patterns have been discarded from further cluster analysis.

4.11.3 Discretization of fAFLP fingerprint patterns

A classification strategy based on the optimization an information theoretic expression such as stochastic complexity, which is a quantitative criterion for evaluation of the global goodness of a given classification with respect to the given data set, requires that the fAFLP fingerprint patterns of the *Vibrio* isolates are transformed into a binary vector representation. Moreover, when comparing classifications of a given data set produced by different classification methods, one has to take into account the influence of the possible alternative similarity models that are used in combination with these clustering techniques. In our case, we want to compare the classification described by Thompson et al. [103] using Ward's hierarchical clustering algorithm [119] with a classification based on the minimization of stochastic complexity, in order to evaluate the classification method incorporated in the BinClass software package [42] for usage in bacterial taxonomy.

In the paper of Thompson *et al.* [103], Ward's classification algorithm works on a similarity matrix containing similarity values calculated using the first pairwise band matching algorithm (position tolerance $\varepsilon = 0.005$) in combination with the Dice similarity coefficient s_D [25]. The classification based on the minimization of stochastic complexity, on the other hand, works upon a binary vector representation of the given data set. Therefore,

we need to apply a multiple band matching algorithm to transform the band representation of the original data set into binary vector representation.

In order to minimize the effect of this data transformation on the BinClass classification, we evaluated several multiple band matching algorithms by calculating the *cophenetic correlation* given by

$$r = \frac{\sum_{i<j} r_{ij}s_{ij} - \frac{2}{n(n-1)} \sum_{i<j} r_{ij} \sum_{i<j} s_{ij}}{\sqrt{\left[\sum_{i<j} {r_{ij}}^2 - \frac{2}{n(n-1)} \left(\sum_{i<j} r_{ij}\right)^2\right] \left[\sum_{i<j} {s_{ij}}^2 - \frac{2}{n(n-1)} \left(\sum_{i<j} s_{ij}\right)^2\right]}}, \quad (4.38)$$

as a means of measuring the strength of the association between two similarity models [97], where $(r_{ij})_{i,j=0}^{n-1}$ and $(s_{ij})_{i,j=0}^{n-1}$ are the similarity matrix representations of the similarity models under comparison. By careful inspection of the formulas (4.13) and (4.38), it is clear that the cophenetic correlation is nothing else than an extension of Pearson's product moment correlation towards symmetric matrices. The matrix based on s_D as described by Thompson et al. [103] was used for estimating the parameter settings of the different band matching methods (position tolerance ε and vector length n). For each multiple band matching method, Thompson's similarity matrix was compared with the s_D similarity matrices generated for a number of different parameter settings of the given multiple band matching method. The parameter values resulting in a maximal cophenetic correlation were then chosen as the optimal setting for the method. For equal-width discretization, the optimal performance was attained for a vector transformation with $n = 149$ classes (position tolerance $\varepsilon = 0.0067$), while the best histogram-based discretization had a position tolerace setting of 0.005, producing binary vectors of length 319. Sliding window discretization was most conservative for the position tolerance parameter ε set to 0.007 and the resolution parameter δ set to 0.001 (so that n = 994).

All cophenetic correlations between the s_D matrices of the different band matching algorithms presented in sections 4.6 and 4.7, applied upon the *Vibrio*/AFLP data set according to the implementations of the Dice coefficient given in Table 4.4, are shown in Figure 4.16. A position tolerance setting of $\varepsilon = 0.005$ was used throughout all pairwise band matching calculations. From this matrix of congruences between the different similarity models we conclude that sliding window discretization transforms the original data representation into binary vector representation with least change of the similarity model (cophenetic correlation 0.944), while the histogram–based method results in much larger change of the similarity model (cophenetic correlation 0.857). It is also remarkable that the sliding window discretization method is highly related to the simple pairwise band matching method (cophenetic correlation 0.971) for this data set, as can be seen from the single linkage classification of the congruence matrix in Figure 4.16. Sliding window discretization can thus be regarded as the multiple alignment counterpart of simple pairwise band matching. Strikingly, both methods were introduced as new band matching in this chapter. Because it is important to mimic the same band position tolerance behaviour and reduce the effect of fragment comparison of the fingerprint patterns within the classification process, in order to enable an objective comparison of the different classification strategies applied to the

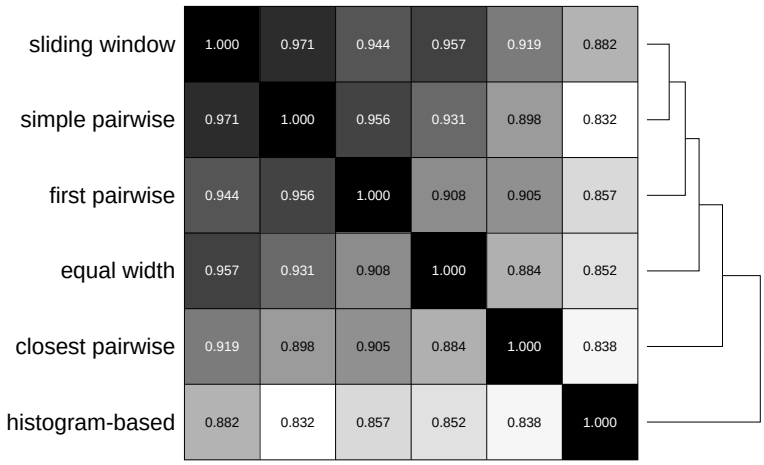


Figure 4.16: Cophenetic correlations between the s_D similarity matrices of different band matching algorithms applied on the *Vibrio*/AFLP data set.

same *Vibrio*/fAFLP data set, the vector transformation produced by the sliding window discretization method was selected for further classification using minimization of stochastic complexity.

In addition to the model congruence matrix shown before, Figure 4.17 depicts the pairwise s_D similarity scatterplots for different similarity models, comparing the original pairwise Dice similarity measurements as calculated by the BioNumerics software package (Applied Maths, Sint-Martens-Latem, Belgium) in the study of Thompson *et al.* [103] plotted along the x-axis, with the corresponding pairwise Dice similarity produced by the band matching methods reviewed in sections 4.6 and 4.7 estimations plotted along the y-axis. A first observation is the quasi perfect correlation ($r = 0.996$) between the first pairwise band matching method and the pairwise band matching results from the BioNumerics software package. This proves that first band matching is indeed the pairwise method implemented in the BioNumerics software. The small deviation of the correlation from the unit value must be due to unavoidable rounding errors during the export procedure of banding patterns and similarity matrices from the BioNumerics software package. A second observation is the apparent relationship between the simple, first and closest pairwise band matching methods given by

$$m_s(B_i, B_j) \geq m_f(B_i, B_j) \geq m_c(B_i, B_j)\,, \tag{4.39}$$

given that the same position tolerance ε is used in all methods. Currently, there is no formal proof of the general validity of this relationship.

4.11.4 Classification of binary vectors

In a first attempt to classify the fAFLP fingerprinting patterns of the *Vibrio* data set, the histogram–based band matching method built into the BioNumerics software package (Applied Maths, Sint-Martens-Latem, Belgium) was used for generating a binary vector

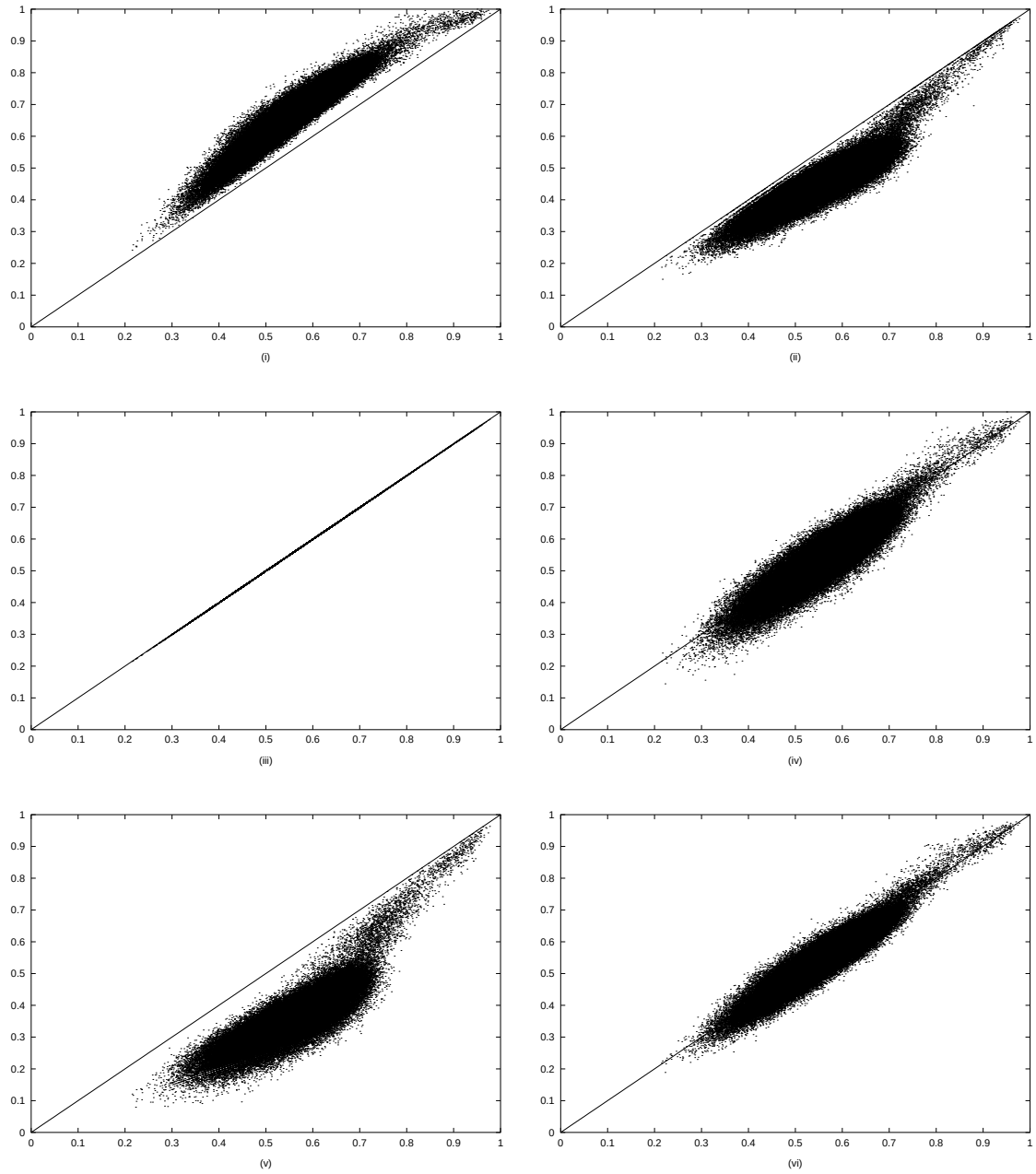


Figure 4.17: Pairwise s_D similarity scatterplots for different similarity models, comparing the original pairwise Dice similarity measurements as calculated by the BioNumerics software package in the study of Thompson *et al.* [103] plotted along the x-axis, with their corresponding pairwise Dice similarity estimations plotted along the y-axis produced by *i*) simple pairwise band matching ($r = 0.956$), *ii*) closest pairwise band matching ($r = 0.905$), *iii*) first pairwise band matching ($r = 0.996$), *iv*) equal-width band matching ($r = 0.908$), *v*) histogram band matching ($r = 0.857$ and *vi*) sliding window discretization ($r = 0.944$).

representation of the AFLP fingerprinting patterns. These binary vectors were then classified by application of the BinClass software package, resulting in the classification as summarized in Table 4.6, which shows only the classes containing type strains. From a taxonomic point of view, this classification was rather unhelpful, as it contains a class (BC1) that harbours many different type strains. Apart from this, the rest of the classification corresponds well with the original classification described by Thompson et al. [103]. These observations have driven us into the exploration of alternative band matching algorithms, presented earlier in this chapter. As a result, we selected the vector representation rendered by the sliding window discretization as a more conservative representation of the information content stored in the original banding patterns.

For a second trial, the binary vectors resulting from the sliding window discretization procedure were classified using minimization of stochastic complexity [40], as it was implemented in the BinClass software package [42]. This algorithm is an example of an unsupervised non-hierarchical classification method, insofar that it does not make use of any prior knowledge or assumptions on the data set other than the binary vector representation of its characteristics and that it presents its final outcome as a plain partitioning of the data set into non-overlapping classes. The default BinClass command line settings were selected, except for the `-F` parameter (safety value) that was set to 50, and the `-S` parameter was set to 20. This has resulted in the classification with 64 classes (labelled BC1,BC2,...,BC64) that is summarized in Table 4.7, by showing only the classes containing type strains. From a taxonomic viewpoint the type strains are now neatly distributed over the different classes, which indicates that this classification is preferable over the previous one. A more detailed presentation of the BinClass classification, showing some group statistics of the different classes and their contained type strains, is given in Table 4.8. The optimal stochastic complexity found for the data set was 739.92. It should be noted that the BinClass software package automatically accommodates to monomorphic bands by discarding vector indexes that have the same binary value before performing the classification, hence taking only into account the bands that are polymorphic within the data set. For reasons of completeness, we have depicted in Figure 4.18 an agglomerative hierarchical clustering built on top of the classification described in Table 4.8. The algorithm used to gradually merge the classes at each agglomerative step was introduced by Gyllenberg et al. [37]. The value at each bifurcation point indicates the stochastic complexity index of the corresponding classification. The leaf nodes of the dendrogram are labelled with the class identifier from the BinClass classification (taxa of the type strains present in each class are indicated between square brackets).

The *Hamming distance* between two binary vectors is defined as the number of bits that are different, from which the distance between two classes can be defined as the mean pairwise Hamming distance between members of the two classes [38]. The *centroid* of a class is by definition the vector giving the frequencies of 1's for the different attributes. By rounding off each component of the centroid to the nearest binary value (0 or 1) one obtains the *hypothetical median organism* (HMO; [64]). The *distortion* of a class, defined as the average number of bits by which the members of the class differ from the HMO (average Hamming distance), can be regarded as a measure of the heterogeneity of a class. *Shannon code length* [20] between a class member and the centroid of the class was used as an

class ID	strain designation	reference no.
BC1	*Salinivibrio costicola*	LMG 11651^T
	Vibrio salmonicida	LMG 14010^T
	Vibrio navarrensis	LMG 15976^T
	Vibrio hollisae	LMG 17719^T
	Vibrio gazogenes	LMG 19540^T
	Photobacterium iliopiscarium	LMG 19543^T
	Vibrio aerogenes	LMG 19650^T
	Vibrio proteolyticus	LMG 3772^T
	Vibrio nigripulchritudo	LMG 3896^T
	Listonella pelagia	LMG 3897^T
	Photobacterium phosphoreum	LMG 4233^T
	Vibrio fischeri	LMG 4414^T
	Vibrio cincinnatiensis	LMG 7891^T
	Vibrio fluvialis	LMG 7894^T
	Vibrio mimicus	LMG 7896^T
	Vibrio orientalis	LMG 7897^T
	Vibrio logei	NCIMB 2252^T
BC2	*Vibrio halioticoli*	LMG 18542^T
BC3	*Vibrio alginolyticus*	LMG 4409^T
BC7	*Vibrio campbellii*	LMG 11216^T
BC12	*Vibrio trachuri*	LMG 19643^T
	Vibrio harveyi	LMG 4044^T
BC13	*Vibrio ichthyoenteri*	LMG 19664^T
BC16	*Vibrio mediterranei*	LMG 11258^T
	Vibrio shiloi	LMG 19703^T
BC19	*Vibrio ordalii*	LMG 13544^T
BC20	*Vibrio splendidus*	LMG 19031^T
BC22	*Vibrio tubiashii*	LMG 10936^T
	Vibrio wodanis	NCIMB 13582^T
BC23	*Vibrio diazotrophicus*	LMG 7893^T
BC25	*Listonella anguillarum*	LMG 4437^T
BC26	*Vibrio vulnificus*	LMG 13545^T
BC27	*Vibrio parahaemolyticus*	LMG 2850^T
BC32	*Vibrio diabolicus*	LMG 19805^T
BC35	*Vibrio pectenicida*	LMG 19642^T
BC36	*Vibrio natriegens*	LMG 10935^T
BC37	*Vibrio furnissii*	LMG 7910^T
BC38	*Vibrio metschnikovii*	LMG 11664^T
BC41	*Vibrio nereis*	LMG 3895^T
BC42	*Vibrio aestuarianus*	LMG 7909^T
BC43	*Vibrio cholerae*	LMG 4406^T
BC44	*Vibrio scophthalmi*	LMG 19158^T
BC47	*Vibrio tapetis*	LMG 19706^T
BC48	*Photobacterium histaminum*	LMG 19445^T
	Photobacterium damselae subsp. *damselae*	LMG 7892^T
BC49	*Vibrio rumoiensis*	LMG 20038^T
	Photobacterium angustum	LMG 8455^T
BC50	*Photobacterium leiognathi*	LMG 4228^T
BC55	*Vibrio penaeicida*	LMG 19663^T
BC57	*Vibrio mytili*	LMG 19157^T

Table 4.6: Distribution of type strains resulting from BinClass classification based on data discretized by the BioNumerics histogram–based band matching method with position tolerance ε set to 0.005. BinClass run resulted in a classification with 61 classes.

class ID	strain designation	reference no.
BC1	*Vibrio halioticoli*	LMG 18542^T
BC2	*Vibrio alginolyticus*	LMG 4409^T
BC5	*Vibrio ordalii*	LMG 13544^T
	Listonella anguillarum	LMG 4437^T
BC7	*Vibrio gazogenes*	LMG 19540^T
	Vibrio fluvialis	LMG 7894^T
	Vibrio furnissii	LMG 7910^T
	Vibrio logei	NCIMB 2252^T
BC11	*Vibrio splendidus*	LMG 19031^T
BC12	*Vibrio trachuri*	LMG 19643^T
	Vibrio harveyi	LMG 4044^T
BC14	*Vibrio ichthyoenteri*	LMG 19664^T
BC18	*Vibrio diabolicus*	LMG 19805^T
BC20	*Photobacterium histaminum*	LMG 19445^T
	Photobacterium damselae subsp. *damselae*	LMG 7892^T
BC21	*Vibrio mediterranei*	LMG 11258^T
	Vibrio shiloi	LMG 19703^T
BC24	*Listonella pelagia*	LMG 3897^T
	Vibrio cincinnatiensis	LMG 7891^T
BC25	*Vibrio proteolyticus*	LMG 3772^T
	Photobacterium phosphoreum	LMG 4233^T
BC26	*Vibrio vulnificus*	LMG 13545^T
BC27	*Vibrio tubiashii*	LMG 10936^T
BC28	*Vibrio parahaemolyticus*	LMG 2850^T
BC29	*Vibrio pectenicida*	LMG 19642^T
BC30	*Vibrio diazotrophicus*	LMG 7893^T
BC33	*Vibrio campbellii*	LMG 11216^T
BC34	*Vibrio salmonicida*	LMG 14010^T
	Photobacterium leiognathi	LMG 4228^T
	Photobacterium angustum	LMG 8455^T
BC37	*Vibrio wodanis*	NCIMB 13582^T
BC38	*Vibrio natriegens*	LMG 10935^T
	Vibrio orientalis	LMG 7897^T
BC39	*Photobacterium iliopiscarium*	LMG 19543^T
	Vibrio fischeri	LMG 4414^T
BC43	*Salinivibrio costicola*	LMG 11651^T
BC44	*Vibrio navarrensis*	LMG 15976^T
	Vibrio hollisae	LMG 17719^T
	Vibrio scophthalmi	LMG 19158^T
BC45	*Vibrio mytili*	LMG 19157^T
BC46	*Vibrio tapetis*	LMG 19706^T
BC48	*Vibrio nereis*	LMG 3895^T
BC49	*Vibrio metschnikovii*	LMG 11664^T
BC50	*Vibrio aestuarianus*	LMG 7909^T
BC52	*Vibrio cholerae*	LMG 4406^T
BC54	*Vibrio rumoiensis*	LMG 20038^T
BC55	*Vibrio aerogenes*	LMG 19650^T
BC57	*Vibrio penaeicida*	LMG 19663^T
BC58	*Vibrio mimicus*	LMG 7896^T
BC62	*Vibrio nigripulchritudo*	LMG 3896^T

Table 4.7: Distribution of type strains resulting from BinClass classification based on data discretized by the sliding window discretization method with ε set to 0.007 and δ set to 0.001 (so that n = 994). BinClass run performed with parameter settings (-F50 -S20), resulting in a classification with 64 classes and a stochastic complexity of 739.92280.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
BC1	24	101 (8)	90-114	A67	*V. halioticoli**	24/24	475	141	BC9	329	BC63	468	206	658
BC2	24	116 (20)	75-147	A49	*V. diabolicus*	2/6	541	171	BC36	312	BC63	467		
				A62	*V. alginolyticus**	22/22							151	494
BC3	22	76 (9)	58-100	A68	***E. norvegicus***	18/18	418	131	BC44	392	BC63	487		
				A69	***E. norvegicus****	4/4							122	412
BC4	21	126 (14)	96-146	A05	***V. neptunius****	21/21	240	78	BC61	257	BC63	488	140	494
BC5	18	93 (13)	60-114	A38	*V. anguillarum**	8/8	523	168	BC42	371	BC63	500	291	759
				A39	*V. ordalii**	10/10							74	366
BC6	16	116 (13)	79-135	A09	***V. fortis****	1/8	482	151	BC11	328	BC63	493	351	1014
				A61	***V. cyclitrophicus***	15/15								
BC7	14	66 (11)	46-91	A11	*V. logei**	1/7	691	243	BC25	356	BC63	491	348	924
				A21	*V. gazogenes**	1/2							325	897
				A27	*V. fluvialis**	4/4							239	697
				A28	*V. fluvialis*	2/2								
				A29	*V. furnisii**	5/5							212	581
				U4	*Vibrio* sp. R-3681	1/1								
BC8	13	134 (19)	106-164	A09	***V. fortis***	5/8	417	133	BC57	340	BC60	478		
				A60	***V. fortis***	8/8								
BC9	13	79 (9)	66-93	A10	*V. pelagius*	1/2	610	206	BC1	329	BC63	483		
				A12	*V. cincinnatiensis*	1/6								
				A64	***V. neonatus****	9/9							167	547
				A65	*Vibrio* sp.	2/2								
BC10	13	120 (12)	105-141	A30	*V. harveyi*	12/14	570	187	BC56	254	BC63	479		
				A59	*V. tubiashii*	1/18								
BC11	12	126 (10)	109-143	A50	*V. splendidus**	11/16	630	218	BC47	322	BC3	463	190	560
				A59	*V. tubiashii*	1/18								
BC12	12	115 (14)	94-147	A36	*V. harveyi**	12/12	520	179	BC6	363	BC63	503	124	435
				A36	*V. trachuri**								208	554
BC13	12	101 (8)	90-112	A66	***V. ezurae****	12/12	388	129	BC2	381	BC63	478	83	321
BC14	12	111 (9)	98-126	A63	*V. ichthyoenteri**	12/12	376	130	BC53	398	BC48	489	248	572
BC15	11	137 (9)	124-149	A04	***V. coralliilyticus***	10/10	271	77	BC16	264	BC58	483		
				A59	*V. tubiashii*	1/18								
BC16	10	134 (9)	116-148	A01	***V. coralliilyticus***	4/10	412	141	BC15	264	BC60	472		
				A02	***V. coralliilyticus***	6/6								
BC17	10	121 (15)	94-149	A52	***V. chagasii***	4/7	588	201	BC36	285	BC63	469		
				A53	***V. chagasii****	5/5							130	470
				A59	*V. tubiashii*	1/18								
BC18	10	109 (15)	72-128	A08	***V. brasiliensis****	7/7	546	175	BC2	368	BC60	473	26	314
				A11	*V. logei*	1/7								
				A49	*V. diabolicus**	2/6							374	836
BC19	9	119 (11)	105-144	A11	*V. logei*	2/7	6	178	BC11	346	BC60	472		
				A55	***V. lentus***	7/7								
BC20	9	116 (17)	95-145	A46	***V. kanaloae****	5/5	621	213	BC36	344	BC63	458	102	440
				A56	*P. ang./P. d. dam.**	2/8							259	661
				A56	*P. histaminum**								292	741
				A59	*V. tubiashii*	2/18								
BC21	9	115 (23)	75-141	A06	*V. med.*/V. shil.*	9/10	468	162	BC55	309	BC58	472	113	423
				A06	*V. med./V. shil.**								208	578
BC22	9	121 (10)	106-143	A45	***V. tasmaniensis****	6/6	461	154	BC47	330	BC25	463	65	308
				A56	*P. ang./P. d. dam.*	1/8								
				A58	*V. pen./V. rum./V. tap.*	2/17								
BC23	9	106 (5)	96-115	A32	*V. harveyi*	4/9	408	137	BC51	271	BC60	469		
				A33	***V. rotiferianus****	5/5							77	324
BC24	8	104 (18)	85-139	A01	***V. coralliilyticus***	4/10	617	213	BC15	290	BC63	497		
				A09	***V. fortis***	2/8								
				A10	*V. pelagius**	1/2							336	903
				A12	*V. cincinnatiensis**	1/6							340	838
BC25	8	64 (13)	50-97	A12	*V. cincinnatiensis*	1/6	599	217	BC10	329	BC63	502		
				A19	*P. phosphoreum**	1/2							341	916
				A25	*V. proteolyticus**	3/3							248	568
				A31	*V. harveyi*	3/3								
BC26	8	85 (9)	73-98	A11	*V. logei*	3/7	498	174	BC20	363	BC63	474		
				A24	*V. vulnificus**	5/5							179	481

Table 4.8: BinClass classification based on data discretized by the sliding window discretization method with the position tolerance parameter ε set to 0.007 and the resolution of the method δ set to 0.001 (so that the vector length d = 994). BinClass run performed with command line settings (-F50 -S20), resulting in a classification with 64 classes and a stochastic complexity of 739.92280. [1]Class identifier, [2]Size (number of strains n), [3]Average number of bands (standard deviation) over all profiles in the class, [4]Minimal and maximal number of bands of all profiles in the class, [5]fAFLP cluster in classification of Thompson et al. [103], [6]fAFLP cluster name as given in Thompson et al. [103]; * indicates position of type strain; bold face indicates revised name since publication of the paper by Thompson *et al.* [103], [7]Frequency of original fAFLP cluster within class, [8]Average Shannon code length of the class, [9]Class distortion, [10]Nearest class, [11]Hamming distance to nearest class, [12]Farthest class, [13]Hamming distance to farthest class, [14]Hamming distance between type strain and hypothetical median organism, [15]Shannon code length of type strain.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
BC27	8	103 (9)	92-119	A12	*V. cincinnatiensis*	1/6	658	231	BC32	294	BC3	460		
				A14	*V. campbellii*	1/4								
				A37	*V. campbellii*	4/10								
				A57	*V. diazotrophicus*	1/4								
				A59	*V. tubiashii**	1/18							275	717
BC28	7	109 (13)	96-128	A42	*V. parahaemolyticus**	6/6	535	182	BC2	362	BC63	488	103	393
				A58	*V. pen./V. rum./V. tap.*	1/17								
BC29	7	112 (17)	84-129	A13	*V. nig./V. ori.*	1/7	419	128	BC40	362	BC63	475		
				A19	*P. phosphoreum*	1/2								
				A43	*V. pectenicida**	5/5							54	288
BC30	7	96 (12)	85-120	A12	*V. cincinnatiensis*	1/6	554	193	BC27	370	BC63	481		
				A34	*V. diazotrophicus*	5/5								
				A57	*V. diazotrophicus**	1/4							247	632
BC31	7	115 (7)	103-122	A03	***V. coralliilyticus***	7/7	158	56	BC16	288	BC60	500		
BC32	7	95 (12)	79-117	A18	*V. navarrensis*	1/2	514	167	BC27	294	BC63	477		
				A37	*V. campbellii*	6/10								
BC33	6	115 (16)	84-131	A14	*V. campbellii**	3/4	598	224	BC10	303	BC3	471	191	484
				A32	*V. harveyi*	1/9								
				A59	*V. tubiashii*	2/18								
BC34	6	115 (15)	101-146	A21	*V. gaz./V. sal.**	1/2	705	253	BC61	380	BC63	479	364	896
				A54	*V. myt./P. lei**	3/5							156	523
				A56	*P. ang.*/P. d. dam.*	1/8							302	778
				A59	*V. tubiashii*	1/18								
BC35	6	123 (17)	103-146	A16	***V. hispanicus****	3/3	559	197	BC17	332	BC3	474	153	461
				A58	*V. pen./V. rum./V. tap.*	1/17								
				A59	*V. tubiashii*	2/18								
BC36	6	124 (12)	107-141	A49	*V. diabolicus*	1/6	663	237	BC17	285	BC60	458		
				A52	***V. chagasii***	3/7								
				A59	*V. tubiashii*	2/18								
BC37	6	91 (15)	66-111	A07	*V. wodanis**	6/6	374	129	BC51	365	BC3	485	127	384
BC38	6	93 (15)	67-109	A13	*V. nig./V. ori.**	1/7	496	170	BC20	362	BC63	497	310	758
				A48	*V. natriegens**	5/5							142	453
BC39	5	98 (20)	66-118	A15	*V. fis.*/P. ili.*	4/4	652	234	BC19	370	BC63	481	129	523
				A15	*V. fis./P. ili.**								270	715
				A56	*P. ang./P. d. dam.*	1/8								
BC40	5	133 (13)	111-147	A51	***V. pomeroyi****	4/6	480	162	BC42	345	BC63	495	175	458
				A58	*V. pen./V. rum./V. tap.*	1/17								
BC41	5	94 (30)	67-149	A12	*V. cincinnatiensis*	1/6	520	175	BC62	318	BC48	454		
				A26	***V. hepatarius****	3/4							28	325
				A59	*V. tubiashii*	1/18								
BC42	5	131 (12)	117-147	A51	***V. pomeroyi***	2/6	626	216	BC40	345	BC3	476		
				A58	*V. pen./V. rum./V. tap.*	2/17								
				A59	*V. tubiashii*	1/18								
BC43	5	76 (12)	61-94	A22	*S. costicola**	2/2	573	209	BC41	380	BC63	475	266	568
				A23	***V. xuii****	3/3							143	534
BC44	5	81 (17)	53-98	A17	*V. scophthalmi**	3/3	611	216	BC32	355	BC63	496	72	430
				A18	*V. navarrensis**	1/2							365	791
				U2	*V. hollisae**	1/1							388	819
BC45	5	103 (13)	83-120	A47	***V. pacinii****	3/3	503	182	BC20	368	BC3	475	109	441
				A54	*V. mytili**	2/5							274	560
BC46	5	145 (9)	135-156	A58	*V. pen./V. rum./V. tap.**	5/17	492	168	BC47	327	BC7	490	42	337
BC47	5	119 (13)	100-136	A50	*V. splendidus*	5/16	440	153	BC11	322	BC13	469		
BC48	4	100 (8)	90-110	A44	*V. nereis**	4/4	425	167	BC61	383	BC14	489	219	514
BC49	4	102 (23)	71-130	A20	*V. metschnikovii**	3/3	353	130	BC5	376	BC63	504	98	255
				A59	*V. tubiashii*	1/18								
BC50	4	96 (6)	87-102	A35	*V. aestuarianus**	4/4	321	115	BC30	374	BC63	490	166	400
BC51	4	99 (14)	84-120	A32	*V. harveyi*	4/9	387	149	BC23	271	BC49	477		
BC52	4	104 (13)	82-113	A40	*V. cholerae**	4/4	489	181	BC56	383	BC63	490	229	594
BC53	3	127 (6)	121-135	A01	***V. coralliilyticus***	2/10	414	150	BC15	272	BC48	471		
				A13	*V. nig./V. ori.*	1/7								
BC54	3	126 (6)	117-131	A58	*V. pen./V. rum.*/V. tap.*	3/17	406	147	BC2	355	BC63	475	71	330
BC55	3	68 (3)	64-71	A06	*V. med./V. shil.*	1/10	519	188	BC21	309	BC63	522		
				A49	*V. diabolicus*	1/6								
				U1	*V. aerogenes**	1/1							181	512
BC56	3	111 (11)	96-122	A13	*V. nig./V. ori.*	1/7	444	161	BC10	254	BC48	476		
				A30	*V. harveyi*	2/14								
BC57	3	128 (22)	98-150	A13	*V. nig./V. ori.*	1/7	508	184	BC8	340	BC59	472		
				A58	*V. pen.*/V. rum./V. tap.*	2/17							210	534
BC58	2	103 (5)	98-107	A41	*V. mimicus**	2/2	257	128	BC27	387	BC63	500	99	257
BC59	2	85 (33)	52-118	A56	*V. tubiashii*	1/8	504	252	BC20	356	BC48	481		
				U3	*Vibrio* sp. R-1586	1/1								
BC60	2	103 (9)	94-111	A57	*V. diazotrophicus*	2/4	218	109	BC13	385	BC31	500		
BC61	2	109 (10)	99-118	A56	*P. d. damselae*	2/8	159	79	BC4	257	BC63	495		
BC62	2	75 (11)	64-86	A13	*V. nig.*/V. ori.*	1/7	390	195	BC41	318	BC63	490	148	390
				A26	***V. hepatarius***	1/4								
BC63	1	127 (0)	127-127	A59	*V. tubiashii*	1/18	1	0	BC16	423	BC55	522		
BC64	1	100 (0)	100-100	A13	*V. nig./V. ori.*	1/7	1	0	BC38	374	BC59	471		

Table 4.9: Continuation of Table 4.8

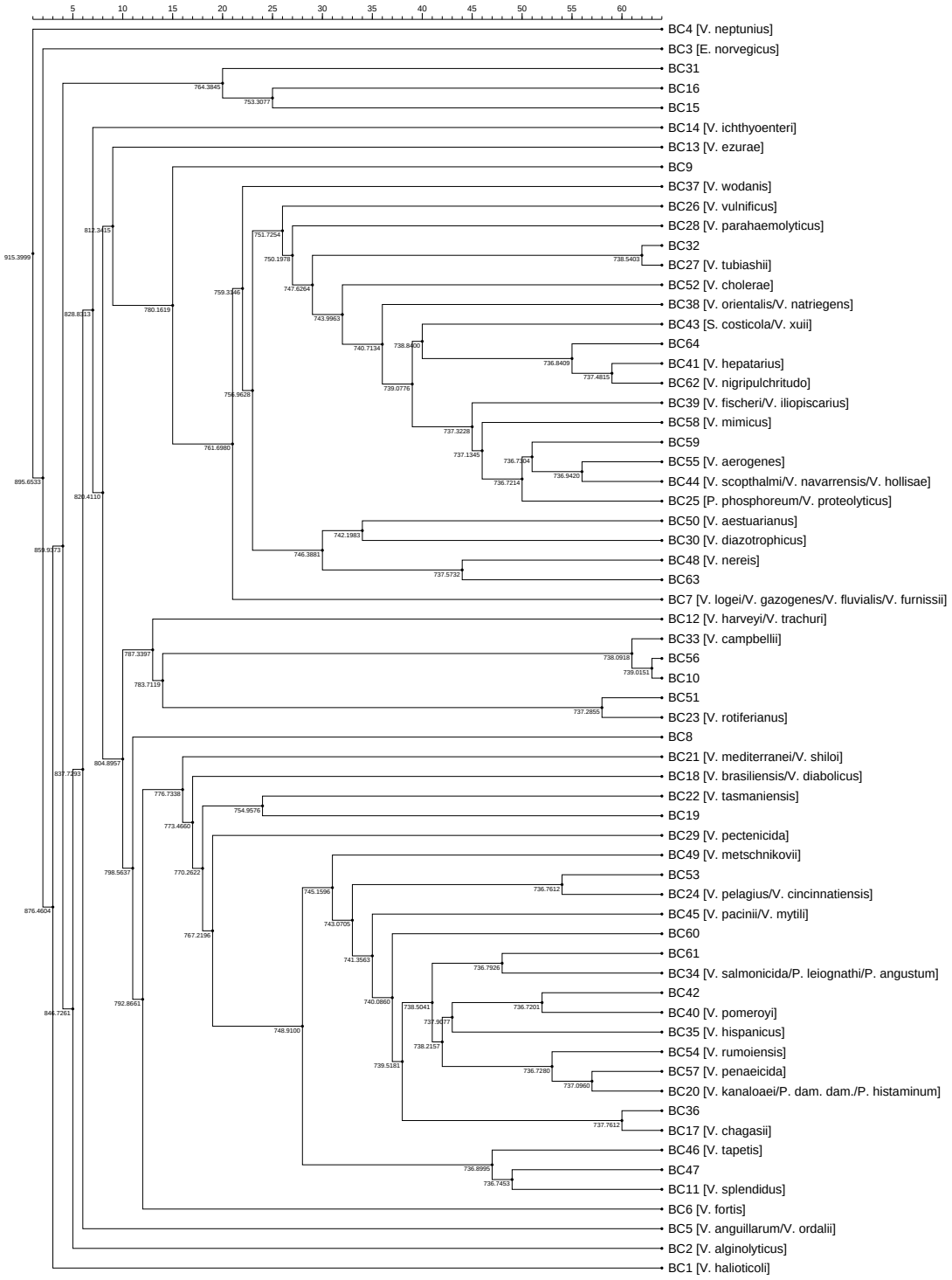


Figure 4.18: Agglomerative hierarchical clustering built on top of the classification described in Table 4.8. The algorithm used to gradually merge the classes at each agglomerative step was introduced by Gyllenberg et al. [37]. The value at each bifurcation point indicates the stochastic complexity index of the corresponding classification. Dendrogram leaf nodes are labelled with the class identifier from the BinClass classification (taxa of the type strains present in each class are indicated between square brackets; *E.* ≡ *Enterovibrio*, *P.* ≡ *Photobacterium*, *S.* ≡ *Salinivibrio*, *V.* ≡ *Vibrio*).

alternative distance function, where the average Shannon code length of all class members with respect to the centroid of the class gives an alternative quantifier for describing the heterogeneity of the class. For both the distortion and the average Shannon code length it holds that classes with lower values for these parameters are more homogeneous than classes with higher values.

From a taxonomic viewpoint the fingerprint profiles of the type strains that were included in the data set are more or less neatly distributed over the different classes that result from the BinClass classification (see Table 4.8), although this was not forced by any subjective descision making within the classification strategy and the type strain information was not regarded as prior knowledge by the classification scheme. This proves that the partitioning of the fAFLP patterns from the current study by minimization of the stochastic complexity, generated classes that generally correspond well with the species delineated within the family *Vibrionaceae*. In this context, both the centroid and the HMO can be regarded as estimations of fictitious representatives for each class, thus by extension also to the species that are represented by these classes. The Hamming distance to the HMO and the Shannon code length may then be employed as measures for evaluating the typicality of a pattern for the class it belongs to, or stated differently, they measure how typical a strain is for the species to which it is identified to by a given classification procedure applied on a chosen set of characters of the strain. As a type strain has to be designated when a species is first described and named, the nomenclatural type strain is nothing more than the name bearer of the species and is usually the first strain known [16]. Hence, at the time of type strain selection so little information has yet been found out about the constellation of the species that will be represented by that strain, that there is not enough statistical evidence in order to assure that the type strain is indeed also a typical strain [98]. Moreover, some of the type strains might be comparatively old and have lost useful characters due to gene loss caused by the long preservation time of the strains.

The source code and a user manual of the BinClass software package are distributed as supplementary data with the online version of the paper [23] that presents part of the results presented in this chapter (see `http://ijs.sgmjournals.org`). Due to the command line interface and the ANSI C compliance of the source code, the software easily compiles on most operating systems (Win32, UNIX, Linux). For convenience of the readership, all BinClass–formatted input files and the output files generated by the software package in the framework of this study were included as well as supplementary data, together with an executable version of the program that has been compiled to run on all Win32 platforms. One of the output files generated by the BinClass software package contains a complete description of the classification results, in which each of the 507 *Vibrionaceae* strains is accounted for. This information is far too extensive to be reported in detail within this chapter.

4.11.5 Comparison of the alternative classifications

In order to rate the value of minimizing stochastic complexity for the classification of bacterial genotyping fingerprint patterns, in this subsection the BinClass classification of the fAFLP patterns generated from the *Vibrionaceae* strains included in the case study is compared with the classification of the same data set as described by Thompson *et al.* [103]. A simple test statistic for measuring the similarity of different classification is provided by the *Rand statistic* [83]. This statistic is defined as the fraction of agreement, i.e. the number of pairs of objects that are either in the same groups in both partitions or in different groups in both partitions or in different groups in both partitions, divided by the total number of pairs of objects. As such, it measures the proportion of consistent allocations by the two classifications. For two classifications C_1 and C_2 of a given set of n items, a formal expression of the Rand statistic is given by

$$\text{Rand}(C_1, C_2) = \frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{j=i+1}^{n} c_{ij} \,, \tag{4.40}$$

where the value c_{ij} is 1 if the two items i and j either belong to the same or to different classes in both classifications, otherwise it equals to zero. The Rand statistic lies between 0 and 1, where 1 indicates that the two partitions are perfectly congruent. If the Rand index is 0, then there is no correspondence between the two classifications. The Rand statistic between the two classifications of the case study is 0.98353, indicating that there is good overall agreement between the two classifications.

A more detailed representation of the congruence between two classifications can be given by the construction of a simple *contigency table*, to depict the cross-classification of items in the two partitions [2]. Figure 4.19 shows a graphical representation of the contigency table for the two classifications compared in this subsection. In this representation, each row represents a class from the classification described previously by Thompson *et al.* [103], with the assigned class identifier in the first column and the number of strains in the last column. Each column represents a class from the BinClass classification, with the assigned class identifier in the first row and the number of strains in the last row. The values in the row–column intersections represent the number of strains that the two corresponding classes have in common. Through manual permutation of the rows and columns of the contigency table, we have transformed the original table into a representation that supports the intuitive appreciation that the more congruence there is between the two classifications, the better the contigency table can be aligned around the diagonal.

As a final inspection of the concordances and disparities between both classifications investigated in this section, the new classification was presented for evaluation by an expert in the taxonomy of Vibrionaceae, being the author of the original hierarchical classification of the fAFLP patterns [103]. The following subsection presents the outcome of this investigation, together with the taxonomic implications for the *Vibrio* data set.

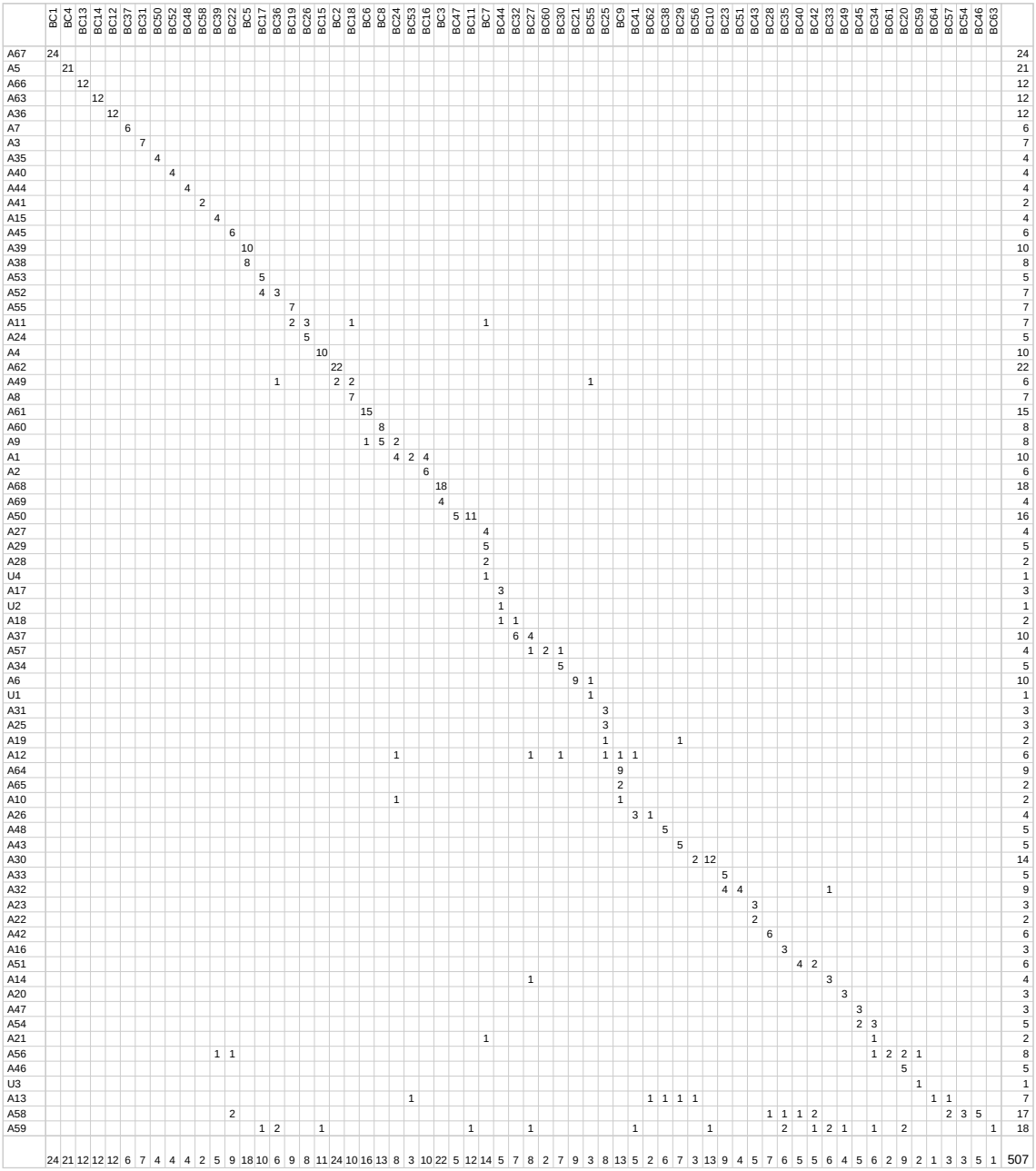


Figure 4.19: Comparison of the classification described by Thompson et al. [103] and the BinClass classification based on data discretized by the sliding window method with the position tolerance parameter ε set to 0.007 and the resolution of the method δ set to 0.001 (so that the vector length d = 994). BinClass run performed with command line settings (-F50 -S20), resulting in a classification with 64 classes and a stochastic complexity of 739.92280. Each row represents a class from the classification described previously by Thompson *et al.* [103], with the assigned class identifier in the first column and the number of strains in the last column. Each column represents a class from the BinClass classification, with the assigned class identifier in the first row and the number of strains in the last row. The values in the row–column intersections represent the number of strains that the two corresponding classes have in common.

4.11.6 Evaluation of classification by domain expert

The 507 strains examined in this study formed 64 classes (BC1,...,BC64), several of which (i.e. BC1, BC4, BC12, BC13, BC14, BC31, BC37, BC48, BC50, BC52, BC58) corresponded exactly with classes of the clustering of the same data set using s_D/Ward and an arbitrary cluster cut off value of 45% [103]. In addition, new relationships among former fAFLP clusters have been disclosed, many of which are in agreement with recent DNA–DNA hybridisation and 16S rDNA sequence experiments. Class BC1 harboured 24 *Vibrio halioticoli* strains, with low pattern distortion ($\approx$ heterogeneity) and was most closely related to BC9 which comprised a new *Vibrio* species, *V. neonatus*, phylogenetically related to *V. halioticoli* [91, 92]. Class BC2 had 24 strains, including the type strain of *V. alginolyticus* and two *V. diabolicus* strains according to the clustering obtained by Thompson et al. [103]. Interestingly, these two *V. diabolicus* strains were originally identified by Vandenberghe et al. [114], using phenotypic and genotypic techniques, as *V. alginolyticus*. The nearest class of BC2 was BC36 which harboured one *V. diabolicus* strain.

Class BC3 harboured 22 *Enterovibrio norvegicus* strains. The combination of the two fAFLP clusters, A68 and A69, into BC3 is in complete agreement with more recent analyses based on DNA hybridisation and 16S rDNA sequences which proved both fAFLP clusters A68 and A69 are in fact a single species, *E. norvegicus* [104]. The nearest class of BC3 was BC44 which holds *V. hollisae*. This is also the closest phylogenetic neighbour of *Enterovibrio* based on 16S rDNA analysis, having about 95% sequence similarity [104]. Class BC4 consisted of 21 *V. neptunius* strains of remarkable low pattern diversity, while class BC5 merged all *V. anguillarum* and *V. ordalii* strains analysed. *V. ordalii* was described by Schiewe et al. [94] to encompass biotype two of *V. anguillarum*. It is well known that these species are highly related, having nearly 100% 16S rDNA similarity and 70% DNA-DNA similarity. Class BC6 hosted the type strain of *V. fortis* and 15 *V. cyclitrophicus* strains, while most of the other members of *V. fortis* appeared in BC8 and BC24. *V. fortis* was proposed to encompass strains of the former fAFLP clusters A9 and A60 [105]. The fact that the type strain of this species clusters apart from all other species members may suggest that the type strain of *V. fortis* is a species on its own, and that this heterogeneous species may be split into new species in the future. Class BC8 merged 5 strains of the former fAFLP group A9 and all 8 strains of A60 which correspond to the newly described *V. fortis*. Clearly BC8 differs from the type strain of *V. fortis* (see BC6).

Class BC7 consisted of 14 strains including the type strains of *V. gazogenes*, *V. fluvialis*, *V. furnissii* and *V. logei*. A remarkable feature of this diverse class is the high distance of all type strains towards the hypothetical median organism (HMO, [64]). It is reasonable that *V. fluvialis* and *V. furnissii* (former biotype of *V. fluvialis*) group together although one would not expect the attraction of the psychrophilic *V. logei* to this class. A 16S rDNA based phylogenetically analysis of *V. logei* revealed that this organism is more related to the psychrophilic vibrios (e.g. *V. fischeri*, *V. salmonicida*, *V. wodanis*) than to any of the species within BC7. Class BC9 (n=13) hosted 9 strains of a new *Vibrio* species, *V. neonatus* [92]. This class attracted two other vibrios named *V. pelagius* and *V. cincinnatiensis* which are most probably representatives of this new species. Classes BC10, BC12, BC23, BC25, BC33, BC51 and BC56 hosted *V. harveyi* strains, although the type strain of *V. harveyi* was

in BC12 suggesting that this is a very diverse species. Whereas BC11 (n=12) harboured the type strain and most reference strains of *V. splendidus* and a strain of *V. tubiashii*. This class was related to BC47. Strains originally allocated to *V. tubiashii* [103] were repartitioned into different BC classes (BC10, BC11, BC15, BC17, BC20, BC27, BC33, BC34, BC35, BC36, BC41, BC42, BC49, BC58 and BC63), being the type strain allocated to BC27. This suggests that the original *V. tubiashii* group as delineated by Thompson et al. [103] was quite artificial. The remaining *V. splendidus* strains formed a separated group, BC47, which may be a variant of *V. splendidus* or a new species. Two strains of BC47 were found in the so called ribotype cluster C described by Mácian et al. [66].

Class BC13 hosted a new *Vibrio* species, *V. ezurae* [91, 92], while BC14 harboured 12 *V. ichthyoenteri* strains. Classes BC15 and BC16 comprised most *V. coralliilyticus* strains [10], although BC24, BC31 and BC53 harboured 4, 7, and 2 strains of this species. It is quite remarkable that BC31 makes a (homogeneous) cluster on its own, suggesting it is a variant of *V. coralliilyticus*. Strains of this class appear to be specialised in causing disease in *Nodipecten nodosus* bivalve larvae, while other *V. coralliilyticus* are known coral pathogens [10]. Class BC17 consisted of 10 strains, including the type strain of *V. chagasii* and most reference strains of this new species [107]. The nearest neighbour of BC17 was BC36 which hosted three *V. chagasii* strains (LMG 13220, LMG 13222, LMG 13239), suggesting that they may be yet another new species. In fact a closer examination of the fAFLP patterns of *V. chagasii* strains and the DNA-DNA hybridisation data indicates a large diversity within this species in support with the new grouping obtained here. In class BC18 two species, *V. brasiliensis* and *V. diabolicus*, were merged. Of course this is not an ideal situation but, the grouping might be just a reflection of the phylogenetic relatedness between the two species as they share about 98% 16S rDNA similarity. BC19 harboured all *V. lentus* strains and two *V. logei* strains. Interestingly, the nearest neighbour class of BC19 was BC11. It is well known that *V. lentus* and *V. splendidus* are highly related species. The strains allocated to *V. logei* in the previous fAFLP analysis [103] were quite heterogeneous, as can be seen with the new partitioning of the strains. The type strain of *V. logei* appeared in BC7, but other 1, 2, and 3 strains appeared in BC18, BC19 and BC26, respectively. These strains were supposed to be *V. logei*, however the results presented here undermine this assumption.

BC20 merged together three type strains (i.e. *P. damselae*, *P. histaminum* and *V. kanaloae*), which might undermine the value of the new classification. However, *P. damselae* was originally clustered with *P. angustum* [103]. *P. angustum* appears now in BC34 which merges the type strains of *V. salmonicida* and *P. leiognathi*. BC21 consists of nine *V. mediterranei* strains, including the former type strain of *V. shilonii* [103]. BC22 comprised six strains of a newly described species, *V. tasmaniensis* [106] which attracted 3 strains from two very heterogeneous fAFLP clusters, A56 and A58. Thompson et al. [103] highlighted that the precise taxonomic allocation of isolates clustering with more than one type strain was unclear, requiring further investigation. Here we demonstrate the partitioning of such isolates by using minimization of stochastic complexity, pointing out to the usefulness of this new approach. BC23 harboured *V. rotiferianus* [31] and four *V. harveyi* strains. According to Gomez-Gil et al. [31] both species are highly related, and the results presented here may suggest that those four strains identified as *V. harveyi* are in fact *V. rotiferianus*. BC23 was

related to BC51 which hosted four diverse *V. harveyi* isolates. Class BC24 attracted the type strains of *V. cincinnatiensis* and *V. pelagius*, whereas BC25 attracted *P. phosphoreum* and *V. proteolyticus*. All these species were clearly separated in the clustering of Thompson et al. [103], but with the SC-minimizing classification one may expect that certain species be grouped together. This fact may be just a reflection of the limitation of band patterns which happen to give similar fingerprints between completely unrelated species (e.g. *P. phosphoreum* and *V. proteolyticus*). We inspected the original patterns of the species within BC25 and found large gaps, which will turn out in zeros in the binarized patterns used for comparisons.

Class BC26 consisted of five *V. vulnificus* strains. This class also attracted three so called *V. logei* strains. Two of this strains i.e. VIB 523 and STD3-996 are clearly *V. vulnificus* representatives misidentified by Thompson et al. [103]. Arias et al. [4] identified VIB 523 to the species *V. vulnificus* using phenotypic and genotypic techniques, while our 16S rDNA sequence of STD3-996 revealed 100% similarity towards *V. vulnificus*. BC27 consisted of 8 strains including the type strain of *V. tubiashii*, while BC28 had 7 strains including the type strain of *V. parahaemolyticus*. BC28 was closely related to BC2 which hosted *V. alginolyticus* (a former variant of *V. parahaemolyticus*). BC29 comprised the species *V. pectenicida* and two other strains which have been probably misidentified in the former analysis of Thompson et al. [103]. Surprisingly, Class BC30 put together the former fAFLP cluster A34 and the type strain of *V. diazotrophicus*. Originally A34 was thought to be a new species, however recent DNA-DNA hybridisation data has proven that A34 belongs to *V. diazotrophicus*. Class BC33 harboured the six strains including the type strain of *V. campbellii*. Whereas BC34 clustered together the type strains of *V. salmonicida*, *P. angustum* and *P. leiognathi*.

Class BC35 accommodated *V. hispanicus* and three other strains. BC38 merged the type strains of *V. orientalis* and *V. natriegens*. Originally the type strains of *V. orientalis* and *V. nigripulchritudo* grouped together with other five strains of uncertain taxonomic position [103]. BC39 grouped *V. fischeri* and *P. iliopiscarius* and so did our previous clustering [103]. BC40 consisted of four *V. pomeroyi* strains, including the type strain, but two strains of this species were found in BC42. This is interesting in that these two strains (LMG 21351 and LMG 21352) differ in fAFLP patterns and the DNA-DNA hybridisation data, suggesting that these two strains are in fact at the outskirts of the species *V. pomeroyi* [107]. Classes BC43 to BC45 suffered all from the same problem, namely merging of more the one type strain. Oddly enough, *Salinivibrio costicola* and *V. xuii* such different species were attracted to the same class, BC43. On the other hand, BC46 consisted of three *V. tapetis* strains which were previously merged with *V. penaeicida* and *V. rumoiensis* in the so called fAFLP cluster A58. In the present analysis *V. rumoiensis* appeared in BC54 and *V. penaeicida* in BC57. Classes BC48, BC49, BC50, BC52 and BC58 harboured *V. nereis*, *V. metschnikovii*, *V. aestuarianus*, *V. cholerae*, and *V. mimicus*, respectively. All these species had previously formed clusters on their own [103].

For the *Vibrio*/AFLP data set, there was good overall agreement between the classification based on the minimization of stochastic complexity and the classification as described by Thompson et al. [103] based on Ward's hierarchical clustering algorithm. However, it

is quite clear that certain heterogeneous fAFLP groups e.g. A12, A13, A56, A58 and 59 were totally repartitioned with the new classification. Obviously, many of the repartitioned strains had been given only tentative names. For instance, A59 harboured the type strain of *V. tubiashii* and thus we assumed that all strains clustering together with this type strain at the level of as low as 45% similarity would belong to the species *V. tubiashii*. Apparently this assumption goes beyond the discrimination of fAFLP and the mathematical algorithms used for fAFLP pattern analysis. Additionally, in other cases (see e.g. BC46, BC54 and BC57) the new classification has repartitioned type strains which were grouping together in a former analysis [103] in separated classes. Some ”hidden”relationships have been disclosed with the help of the BinClass classification. This is the case for the former fAFLP groups A68 and A69 which appear in class BC3 (*Enterovibrio norvegicus*). Likewise A9 and A60 are merged in BC8 (*Vibrio fortis*), clusters A34 and A57 in BC30 (*V. diazotrophicus*), and also clusters A14 and A37 in BC27 (*V. campbellii*). All these four classes have been validated by DNA–DNA hybridisation data which showed that these classes harbour strains from the same species [104, 105].

4.12 Conclusions and future perspectives

The study dealt with in the current chapter, originated from our general interest to bridge the gap between the genotypic information provided by molecular fingerprint patterns – nowadays intensively used for establishing reliable bacterial taxonomies – and the application of the modern techniques in mathematics and computational science for finding groups in data. Where the polyphasic paradigm has gradually settled down in the field of bacterial taxonomy, stating that a natural classification of the microorganisms should encompass all relationships exposed by a broad range of their genotypic and phenotypic features, the same way of thinking should be extended to the usage of computational methods for analyzing and visualizing the relationships among bacterial strains. Sole reliance on hierarchical clustering techniques, since long the dominant instrument in the taxonomist’s toolbox, might skew the perception of the bacterial features or fail to extract some of the hidden relationships. From the observation that there might be several meaningful groupings to explain a multifaceted set of data, a variety of cluster analysis techniques will be needed to reveal them all.

Applying state-of-the-art classification methods for the analysis of genotyping fingerprint patterns, often includes several transformations of the original data representation into a more workable computational format. In this chapter we have shown that a naive choice of the discretization method for turning molecular banding patterns into binary vector format can have a harmful impact on the final classification of the profiles. This has led us into an evaluation of the existing multiple band matching methods and the introduction of a new technique – called sliding window discretization – for transforming genotypic fingerprinting data into binary vector format. In the context of an extensive set of fAFLP fingerprint patterns from strains of the family *Vibrionaceae*, it was demonstrated that sliding window discretization results in the most lossless vector transformation compared to other methods. Accordingly, the binary vectors were classified based on the minimization

of stochastic complexity, as an alternative strategy for the hierarchical clustering algorithms that are commonly used in bacterial taxonomy. A scrutinized comparison of the classifications for the same set of fAFLP fingerprint patterns by different classification strategies has revealed that there was good overall correspondence between the alternative groupings, but also confirmed that no single classification managed to reflect all the taxonomic relationships within the *Vibrionaceae*.

One of the exiting opportunities offered by successful transformation of electrophoresis patterns into vector representation, in the sense that the transformation does not lead to heavy reduction of the information content originally stored in the fingerprint patterns, is that a wide scope of other well-founded clustering methods are applicable for further analysis of the data: k-means, fuzzy c-means, artificial neural networks (both supervised and unsupervised), support vector machines, among many others. Many of these methods have not yet proven their value within the domain of bacterial taxonomy, and certainly not for the classification of molecular fingerprints. And still the important open question remains: how to intelligently merge all aspects learned from several genotypic and phenotypic bacterial features and how to combine classification results from different clustering methods into a consensus taxonomy of microbial diversity that is stable, descriptive, predictive, objective and highly informative. And preferably this consensus taxonomy should be established in an automated and dynamic way.

Bibliography

[1] **Abramowitz, M. & Stegun, I. A. (1968)**. Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables. Nat Bur of Stand Appl Math Ser, volume 55, 7th printing, US Government Printing Office, Washington, D. C.

[2] **Anderberg, M. R. (1973)**. Cluster Analysis for Applications. Academic Press, New York, NY, USA.

[3] **Apostolico, A. & Giancarlo, R. (1998)**. Sequence alignment in molecular biology. *J Comp Biol* **5**, 173–196.

[4] **Arias, C. R., Verdonck, L., Swings, J., Aznar, R. & Garay, E. (1997).** Intraspecific differentiation of *Vibrio vulnificus* biotypes by amplified fragment length polymorphism and ribotyping. *Appl Environ Microbiol* **63**, 2600–2606.

[5] **Austin, B., Stuckey, L. F., Robertson, P. A. W., Effendi, I. & Griffith, D. R. W. (1995).** A probiotic strain of *Vibrio alginolyticus* effective in reduncing disease caused by *Aeromonas salmonicida*, *Vibrio anguillarum* and *Vibrio ordalii*. *J Fish Dis* **18**, 93–96.

[6] **Austin, B., Alsina, M., Austin, D. A., Blanch, A. R., Grimont, F., Grimont, P. A. D., Jofre, J., Koblavi, S., Larsen, J. L., Pedersen, K., Tiainen, T., Verdonck, L.& Swings, J. (1995).** Identification and typing of *Vibrio anguillarum*: a comparison of different methods. *Syst Appl Microbiol* **18**, 285–302.

[7] **Austin, B., Austin, D. A., Blanch, A. R., Cedra, M., Grimont, F., Grimont, P. A. D., Jofre, J., Koblavi, S., Larsen, J. L., Pedersen, K., Tiainen, T., Verdonck, L.& Swings, J. (1997).** A comparison of methods for the typing of fish-pathogen *Vibrio* spp. *Syst Appl Microbiol* **20**, 89–101.

[8] **Austin, B. & Austin, D. A. (1999).** Bacterial fish pathogens. Disease of farmed and wild fish. Springer Praxis Books, series in Aquaculture and Fisheries, 3rd ed.

[9] **Austin, B., Dawyndt, P., Gyllenberg, M., Koski, T., Lund, T., Swings, J., Thompson, F. L. (2004).** Sliding window discretization: a new method for multiple band matching of bacterial genotyping fingerprints. *Bull Math Biol*, **66(6)**, 1575–1596.

[10] **Ben-Haim, Y., Thompson, F. L., Thompson, C. C., Cnockaert, M. C., Hoste, B., Swings J. & Rosenberg, E. (2002).** *Vibrio coralliilyticus* sp. nov., a temperature-dependent pathogen of the coral *Pocillopora damicornis*. *Int J Syst Evol Microbiol* **53**, 309–315.

[11] **Ball, G. H. & Hall, D. J. (1965).** ISODATA, a novel method of data analysis and pattern classification. Tech. Rep. NTIS No. AD699616, Stanford Research Institute, Menlo Park.

[12] **Baker, F. B. & Hubert, L. J. (1975).** Measuring the power of hierarchical cluster analysis. *Journal of the American Statistical Association* **70**, 31–38.

[13] Bergey's Manual of Systematic Bacteriology, 2nd ed., 2004. `http://dx.doi.org/10.1007/bergeysoutline200310`.

[14] **Brenner, D. J. (1973).** Deoxyribonucleic acid reassociation in the taxonomy of enteric bacteria. *Int J Syst Bacteriol* **23**, 298–307.

[15] **Borg, I. & Groenen, P. (1997).** Modern Multidimensional Scaling. Springer, New York, NY, USA.

[16] **Buchanan, R. E. (1925).** General Systematic Bacteriology. Williams & Wilkins Co., Baltimore, MD, USA.

[17] **Calinski, R. B. & Harabasz, J. (1974).** A dendrite method for cluster analysis. *Communications in Statistics* **3**, 1–27.

[18] **Chan, S. C., Wong, A. K. C. & Chiu, D. K. Y. (1992).** A survey of multiple sequence comparison methods. *Bull Math Biol*, **54**, 563–598.

[19] **Chen, C. Y., Wu, K. M., Chang, Y. C., Chang, C. H., Tsai, H. C., Liao, T. L., Liu, Y. M., Chen, H. J., Shen, A. B., Li, J. C., Su, T. L., Shao, C. P., Lee, C. T., Hor, L. I., Tsai, S. F. (2003).** Comparative genome analysis of *Vibrio vulnificus*, a marine pathogen. *Genome Res* **13**, 2577–2587.

[20] **Cover, T. M. & Thomas, J. A. (1991).** Elements of Information Theory. John Wiley & Sons Inc., New York, NY, USA.

[21] **Cox, T. F. & Cox, M. A. A. (1994).** Multidimensional Scaling. Chapman & Hall, London, UK.

[22] **Davies, D. L. & Bouldin, D. W. (1979).** A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **1**, 224–227.

[23] **Dawyndt, P., Thompson, F. L., Austin, B., Swings, J., Koski, T. & Gyllenberg, M. (in press).** Application of sliding window discretization and minimization of stochastic complexity for the analysis of fAFLP genotyping data of *Vibrionaceae*. *Int J Syst Evol Microbiol*.

[24] **De Baets, B., De Meyer, H. & Naessens, H. (2001).** A class of rational cardinality-based similarity measures. *J Comput Appl Math* **132(1)**, 51–69.

[25] **Dice, L. R. (1945).** Measures of the amount of ecological association between species. *Ecology* **26**, 297–302.

[26] **Eilers, H., Pernthaler, J. & Amann, R. (2000).** Succession of pelagic marine bacteria during enrichment: a close look at cultivation-induced shifts. *Appl Environ Microbiol* **66(11)**, 4634–4640.

[27] **Farmer, J. J. III, Carter, G. P., Miller, V. L., Falkow, S. & Wachsmuth, I. K. (1992).** Pyrazinamidase, CR-MOX agar, salicin fermentation-esculin hydrolysis, and D-xylose fermentation for identifying pathogenic serotypes of *Yersinia enterocolitica*. *J Clin Microbiol* **30(10)**, 2589–2594.

[28] **Feltham, R. K. A. & Sneath, P. H. A. (1979).** Quantitative comparison of electrophoretic traces of bacterial proteins. *Comp Biomed Res* **12**, 247–263.

[29] **Fickett, J. W. (1984).** Fast optimal alignment. *Nucl Acids Res* **12**, 175–180.

[30] **Forbes, K. J., Bruce, K. D., Jordens, J. Z., Ball, A. & Pennington, T. H. (1991).** Rapid methods in bacterial DNA fingerprinting. *J Gen Microbiol* **137**, 2051–2058.

[31] **Gomez-Gil, B., Thompson, F. L., Thompson, C. C. & Swings, J. (2003).** *Vibrio pacinii* sp. nov., a bacterium isolated from reared aquatic organisms. *Int J Syst Evol Microbiol* **53**, 1569–1573.

[32] **Gomez-Gil, B., Thompson, F. L., Thompson, C. C., Garcia-Gasca, A., Roque, A. & Swings, J. (2004).** *Vibrio hispanicus* sp. nov., isolated from *Artemia* sp. and sea water in Spain. *Int J Syst Evol Microbiol* **54(1)**, 261–265.

[33] **Gray, R. M.& Gersho, A. (1991).** Vector quantization and signal compression. Kluwer Academic Publishers.

[34] **Gotch, O. (1982).** An improved algorithm for matching biological sequences. *J Mol Biol* **162**, 705–708.

[35] **Gusfield, D. (1997).** Algorithms on Strings, Trees, and Sequences. Cambridge University Press, New York, NY, USA.

[36] **Gyllenberg, H. G., Gyllenberg, M., Koski, T., Lund, T., Schindler, J. & Verlaan, M. (1997).** Classification of *Enterobacteriaceae* by minimization of stochastic complexity. *Microbiology* **143**, 721–732.

[37] **Gyllenberg, H. G., Gyllenberg, M., Koski, T. & Lund, T. (1998).** Stochastic complexity as a taxonomic tool. *Computer Methods and Programs in Biomedicine* **56**, 11–22.

[38] **Gyllenberg, H. G., Gyllenberg, M., Koski, T., Lund, T. & Schindler, J. (1999).** *Enterobacteriaceae* taxonomy approached by minimization of stochastic complexity. *Quantitative Microbiology* **1**, 157–170.

[39] **Gyllenberg, M. & Koski, T. (1996).** Numerical taxonomy and the principle of maximum entropy. *Journal of Classification* **13**, 213–229.

[40] **Gyllenberg, M., Koski T. & Verlaan, M. (1997).** Classification of binary vectors by stochastic complexity. *J Multivariate Anal* **63**, 47–72.

[41] **Gyllenberg, M. & Koski, T. (2001).** Probabilistic models for bacterial taxonomy. *International Statistical Review* **69**, 249–276.

[42] **Gyllenberg, M., Koski T. & Lund, T. (2001).** BinClass: a software package for classifying binary vectors. User's guide. *TUCS Technical Report* **411**. `http://www.tucs.fi/publications/techreports/TR411.php`.

[43] **Gyllenberg, M., Dawyndt, P., Koski, T., Lund, T., Thompson, F., Austin, B. & Swings, J. (2002).** New methods for the analysis of binarized BIOLOG GN data of *Vibrio* species: minimization of stochastic complexity and cumulative classification. *Syst Appl Microbiol* **25**, 403–415.

[44] **Gyllenberg, M., J. Carlsson, and T. Koski (2003).** Bayesian network classification of binarized DNA fingerprinting patterns. In: V. Capasso (ed.): *Mathematical Modelling and Computing in Biology and Medicine*, Progetto Leonardo, Bologna, 2003, 60–66.

[45] **Hartigan, J. A. (1975).** Clustering Algorithms. John Wiley & Sons, New York, NY, USA.

[46] **Hedlund, B.P. & Staley, J. T. (2001).** *Vibrio cyclotrophicus* sp. nov., a polycyclic aromatic hydrocarbon (PAH)-degrading marine bacterium. *Int J Syst Evol Microbiol* **51(1)**, 61–66.

[47] **Heidelberg, J. F., Eisen, J. A., Nelson, W. C., Clayton, R. A, Gwinn, M. L., Dodson, R. J., Haft, D. H., Hickey, E. K., Peterson, J. D., Umayam, L., Gill, S. R., Nelson, K. E., Read, T. D., Tettelin, H., Richardson, D., Ermolaeva, M. D., Vamathevan, J., Bass, S., Qin, H., Dragoi, I., Sellers, P., McDonald, L., Utterback, T., Fleishmann, R. D., Nierman, W. C. & White, O. (2000).** DNA sequence of both chromosomes of the cholera pathogen *Vibrio cholerae*. *Nature* **406(6795)**, 477–483.

[48] **Hubert, L. J. & Levin, J. R. (1976).** A general statistical framework for assessing categorical clustering in free recall. *Psychological Bulletin* **83**, 1072–1080.

[49] **Hunter, L. (1993).** Molecular biology for computer scientists. In: Hunter, L. (ed.): *Artificial Intelligence and Molecular Biology*, 1–46, AAAI Press Books.

[50] **Huys, G., Coopman, R., Vancanneyt, M., Kersters, I., Verstraete, W., Kersters, K. & Janssen, P. (1993).** High resolution differentiation of Aeromonads. *Medical Microbiology Letters* **2(5)**, 248–255.

[51] **Huys, G. & Swings, J. (1999).** Evaluation of a fluorescent amplified fragment length polymorphism methodology for the genotypic discrimination of *Aeromonas* taxa. *FEMS Microbiol Lett* **177**, 83–92.

[52] **Jaccard, P. (1908).** Nouvelles recherches sur la distribution florale. *Bulletin de la société Vaudoise des Sciences Naturelles* **44**, 223–270.

[53] **Jackman, P. J. H., Feltham, R. K. A. & Sneath, P. H. A. (1983).** A program in BASIC for numerical taxonomy of microorganisms based on electrophoretic protein patterns. *Microb Lett* **23**, 87–89.

[54] **Janssen, P., Coopman, R., Huys, G., Swings, J., Bleeker, M., Vos, P., Zabeau, M. & Kersters, K. (1996).** Evaluation of the DNA fingerprinting method AFLP as a new tool in bacterial taxonomy. *Microbiology* **142**, 1881–1891.

[55] **Kaufman, L. & Rousseeuw, P. (1990).** Finding groups in data: an introduction to cluster analysis. John Wiley & Sons Ltd, New York, NY, USA.

[56] **Kersters, K. & Deley, J. (1975).** Identification and grouping of bacteria by numerical analysis of their electrophoretic protein patterns. *J Gen Microbiol* **87**, 333–342.

[57] **Kim, Y. R., Lee, S. E., Kim, C. M., Kim, S. Y., Shin, E. K., Shin, D. H., Chung, S. S., Choy, H. E., Progulske-Fox, A., Hillman, J. D., Handfield, M., Rhee, J. H. (2003).** Characterization and pathogenic significance of *Vibrio vulnificus* antigens preferentially expressed in septicemic patients. *Infect Immun* **71(10)**, 5461–5471.

[58] **Kita-Tsukamoto, K., Oyaizu, H., Nanba, K. & Simidu, U. (1993).** Phylogenetic relationships of marine bacteria, mainly members of the family *Vibrionaceae*, determined on the basis of 16S rRNA sequences. *Int J Syst Bacteriol* **43(1)**, 8–19.

[59] **Koeleman, J. G., Stoof, J., Biesmans, D. J., Savelkoul, P. H., Vandenbroucke-Grauls, C. M. (1998).** Comparison of amplified ribosomal DNA restriction analysis, random amplified polymorphic DNA analysis, and amplified fragment length polymorphism fingerprinting for identification of *Acinetobacter* genomic species and typing of *Acinetobacter baumannii*. *J Clin Microbiol* **36(9)**, 2522–2529.

[60] **Kohonen, T. (1997).** Self-Organizing Maps, Springer-Verlag, Berlin, Germany, 2nd Edition.

[61] **Krzanowski, W. J. & Lai, Y. T. (1985).** A criterion for determining the number of groups in a data set using sum of squares clustering. *Biometrics* **44**, 23–34.

[62] **Lasters, I., Leyns, F. & Jackman, P. H. J. (1985).** Background estimation in one-dimensional electrophoregrams of whole-cell protein extracts. *Electrophoresis* **6**, 508–511.

[63] **Linde, Y., Buzo, A. & Gray, R. M. (1980).** An algorithm for vector quantizer design. *IEEE Trans Comm* **28**, 84–95.

[64] **Liston, J., Wiebe, W. J. & Colwell, R. R. (1963).** Quantitative approach to the study of bacterial organisms. *J Bacteriol* **85**, 1061–1070.

[65] **Logan, N. A. (1994).** Bacterial Systematics. Blackwell Scientific Publications, Oxford, England.

[66] **Mácian, M. C., Garay, E., Gonzalez–Candelas, F., Pujalte, M. J. & Aznar, R. (2000).** Ribotyping of *Vibrio* populations associated with cultured oysters (*Ostrea edulis*). *Syst Appl Microbiol* **23**, 409–417.

[67] **Makino, K., Oshima, K., Kurokawa, K., Yokoyama, K., Uda, T., Tagomori, K., Iijima, Y., Najima, M., Nakano, M., Yamashita, A., Kubota, Y., Kimura, S., Yasunaga, T., Honda, T., Shinagawa, H., Hattori, M. & Iida, T. (2003).** Genome sequence of *Vibrio parahaemolyticus*: a pathogenic mechanism distinct from that of *V. cholerae*. *Lancet* **361(9359)**, 743–749.

[68] **McClure, M. A., Vasi, T. K. & Fitch, W. M. (1994).** Comparative analysis of multiple protein-sequence alignment methods. *Mol Biol Evol* **11**, 571–592.

[69] **Milligan, G. W. (1981).** A Monte Carlo study of thirty internal criterion measures for cluster analysis. *Multivariate Behavioral Research* **16**, 187–199.

[70] **Milligan, G. W. & Cooper, M. C. (1985).** An examination of procedures for determining the number of clusters in a data set. *Psychometrica* **50**, 159–179.

[71] **Moss, S., Leamaster, B. R. & Sweeney, J. N. (2000).** Relative abundance and species composition of Gram-negative, aerobic bacteria associated with the gut of juvenile white shrimp *Litopenaeus vannamei* reared in oligotrophic well water and eutrophic pond water. *J World Aquac* **31**, 255–263.

[72] **Mougel, C., Thioulouse, J., Perrière, G. & Nesme, X. (2002).** A mathematical method for determining genome divergence and species delineation using AFLP. *Int J Syst Appl Microbiol* **52**, 573–586.

[73] **Needleman, S. B. & Wunsch, C. D. (1970).** A general method applicable to the search for similarities in the amino acid sequences of two proteins. *J Mol Biol* **48**, 443–453.

[74] **Owen, R. J. (1990).** Chromosomal DNA fingerprinting–a new method of species and strain identification applicable to microbial pathogens. *J Med Microbiol* **30**, 89–99.

[75] **Pearson, W. H. (1966).** Estimation of a correlation measure from an uncertainty measure. *Psychometrika* **31(3)**, 421–433.

[76] **Pedersen, K., Verdonck, L., Austin, B., Austin, D. A., Blanch, A. R., Grimont, P. A. D., Jofre, J., Koblavi, S., Larsen, J. L., Tiainen, T., Vigneulle, M. & Swings, J. (1998).** Taxonomic evidence that *Vibrio carchariae* Grimes et al. 1985 is a junior synonym of *Vibrio harveyi* (Johnson and Shunk 1936) Baumann et al. 1981. *Int J Syst Bacteriol* **48**, 749–758.

[77] **Pitcher, D. G., Saunders, N. A. & Owen, R. J. (1989).** Rapid extraction of bacterial genomic DNA with guanidium thiocyanate. *Lett Appl Microbiol* **8(4)**, 151–156.

[78] **Plikaytis, B. D., Plikaytis, B. B. & Shinnick, T. (1992).** Computer-assisted pattern recognition model for the identification of slowly growing mycobacteria including *Mycobacterium tuberculosis*. *J Gen Microbiol* **138**, 2365–2273.

[79] **Pot, B., Vandamme, P. & Kersters, K. (1994).** Analysis of electrophoretic whole-organism protein fingerprints. In: Goodfellow, M. & O'Donnell, A. G. (eds.): *Chemical Methods in Prokaryotic Systematics*, John Wiley and Sons Ltd, Chichester, UK.

[80] **Priest, F. & Austin, B. (1993).** Modern Bacterial Taxonomy. Chapman and Hall, London, Second edition.

[81] **Rademaker, J. L. W., Louws, F. J., Rossbach, U., Vinuesa, P. & de Bruijn, F. J. (1999).** Computer-assisted pattern analysis of genotyping fingerprints and database construction. *Microbial Ecology Manual* **7.1.3**, Kluwer Academic Press.

[82] **Ramaiah, N., Hill, R. T., Chun, J., Ravel, J., Matte, M. H., Straube, W. L. & Colwell, R. R. (2000).** Use of a chiA probe for detection of chitinase genes in bacteria from the Chesapeake Bay. *FEMS Microbiol Ecol* **34(1)**, 63–71.

[83] **Rand, W. M. (1971).** Objective criteria for the evaluation of clustering methods. *J Am Stat Assoc* **66**, 846–850.

[84] **Ratkowsky, D. A. & Lance, G. N. (1978).** A criterion for determining the number of groups in a classification. *Australian Computer Journal* **10**, 115–117.

[85] **Ringo, E. & Birkbeck, T. H. (1999).** Intestinal microflora of fish larvae and fry. *Aquac Res* **30**, 73–93.

[86] **Rissanen, J. (1989).** Stochastic Complexity in Statistical Inquiry. World Scientific, Singapore.

[87] **Rohlf, F. J. (1974).** Methods of comparing classifications. *Annual Review of Ecology and Systematics* **5**, 101–113.

[88] **Rosenberg, E., Ben-Haim, Y., Toren, A., Banin, E., Kushmaro, A., Fine, M. & Loya, Y. (1999).** Effect of temperature on bacterial bleaching of corals. In: Rosenberg, E. (ed.): *Microbial ecology and infectious disease* ASM Press, Washington DC, 242–254.

[89] **Salamon, H., Segal, M., Ponce de Leon, A. & Small, P. M. (1998).** Accomodating error analysis in comparison and clustering of molecular fingerprints. *Emerging Infectious Diseases* **4**, 159–168.

[90] **Sawabe, T., Sugimura, I., Ohtsuka, M., Nakano, K., Tajima, K., Ezura, Y. & Christen, R. (1998).** *Vibrio halioticoli* sp. nov., a non-motile alginolytic marine bacterium isolated from the gut of the abalone *Haliotis discus hannai*. *Int J Syst Bacteriol* **48(2)**, 573–580.

[91] **Sawabe, T., Thompson, F. L., Heyrman, J., Cnockaert, M., Hayashi, K., Tanaka, R., Yoshimizu, M., Hoste, B., Swings J. & Ezura Y. (2002).** Fluorescent amplified fragment length polymorphism and repetitive extragenic palindrome-PCR fingerprinting reveal host-specific genetic diversity of *Vibrio halioticoli*-like strains isolated from the gut of japanese abalone. *Appl Environ Microbiol* **68**, 4140–4144.

[92] **Sawabe, T., Hayashi, K., Moriwaki, J., Thompson, F. L., Swings, J. & Christen R. (2004).** *Vibrio neonatus* sp. nov. and *Vibrio ezurae* sp. nov. isolated from the gut of Japanese abalones. *Syst Appl Microbiol*, **27**, 527-534.

[93] **Sawabe, T., Hayashi, K., Moriwaki, J., Thompson, F. L., Swings, J., Potin, P., Christen, R. & Ezura, Y. (2004).** *Vibrio gallicus* sp. nov., isolated from the gut of the French abalone *Haliotis tuberculata*. *Int J Syst Evol Microbiol* **54(3)**, 843–846.

[94] **Schiewe, M. H., Trust, T. J. & Crosa, J. H. (1981).** *Vibrio ordalii* sp. nov., a causative agent of Vibriosis in fish. *Curr Microbiol* **6**, 343–348.

[95] **Smith, S. K., Sutton, D. C., Fuerst, J. A., Reichelt, J. L. (1991).** Evaluation of the genus *Listonella* and reassignment of *Listonella damsela* (Love et al.) MacDonell and Colwell to the genus *Photobacterium* as *Photobacterium damsela* comb. nov. with an emended description. *Int J Syst Bacteriol* **41(4)**, 529–534.

[96] **Smith, T. F. & Waterman, M. S. (1981).** Identification of common molecular subsequences. *J Mol Biol* **147)**, 195–197.

[97] **Sneath, P. H. A. & Sokal R. R. (1973).** Numerical Taxonomy. The Principles and Practice of Numerical Classification. W. H. Freeman and Co., San Francisco, USA.

[98] **Sneath, P. H. A. (1984).** Bacterial Nomenclature. In: *Bergey's Manual of Systematic Bacteriology.* Williams & Wilkins Co., Baltimore, MD, USA, 19–23.

[99] **Stackebrandt, E., Myrray, R. G. E. & Trüper, H. G. (1988).** Proteobacteria classis nov., a name for the phylogenetic taxon that includes the 'purple bacteria and their relatives'. *Int J Syst Bacteriol* **38**, 321–325.

[100] **Stackebrandt, E., Frederiksen, W., Garrity, G. M., Grimont, P. A., Kampfer, P., Maiden, M. C., Nesme, X., Rossello-Mora, R., Swings, J., Trüper, H. G., Vauterin, L., Ward, A. C. & Whitman, W.B. (2002).** Report of the ad hoc committee for the re-evaluation of the species definition in bacteriology. *Int J Syst Evol Microbiol* **52(3)**, 1043–1047.

[101] The BioNumerics Manual, version 3.0. Applied Maths BVBA, Sint-Martens-Latem, Belgium.

[102] **Thompson, C. C., Thompson, F. L., Vandemeulebroecke, K., Hoste, B., Dawyndt, P. & Swings, J. (2004).** Use of recA as an alternative phylogenetic marker in the family *Vibrionaceae*. *Int J Syst Evol Microbiol* **54(3)**, 919–924.

[103] **Thompson, F. L., Hoste, B., Vandemeulebroecke, K. & Swings, J. (2001).** Genomic diversity amongst *Vibrio* isolates from different sources determined by fluorescent amplified fragment length polymorphism. *Syst Appl Microbiol* **24**, 520–538.

[104] **Thompson, F. L., Hoste, B., Thompson, C. C., Goris, J., Gomez–Gil, B., Huys, L. & Swings, J. (2002).** *Enterovibrio norvegicus* gen. nov., sp. nov., isolated from the gut of turbot (*Scophthalmus maximus*) larvae: a new member of the family *Vibrionaceae*. *Int J Syst Evol Microbiol* **52**, 2015–2022.

[105] **Thompson, F. L., Thompson, C. C., Hoste, B., Vandemeulebroecke, K., Gullian, M. & Swings, J. (2003).** *Vibrio fortis* sp. nov. and *Vibrio hepatarius* sp. nov., isolated from aquatic animals and the marine environment. *Int J Syst Evol Microbiol* **53**, 1495–1501.

[106] **Thompson, F. L., Thompson, C. C. & Swings, J. (2003).** *Vibrio tasmaniensis* sp. nov., isolated from atlantic salmon (*Salmo salar L.*). *Syst Appl Microbiol* **26**, 65–69.

[107] **Thompson, F. L., Thompson, C. C., Li, Y., Gomez–Gil, B., Vandenberghe, J. & Swings, J. (2003).** *Vibrio kanaloae* sp. nov, *Vibrio pomeroyi* sp. nov. and *Vibrio chagasii* sp. nov., from sea water and marine animals. *Int J Syst Evol Microbiol* **53**, 753–759.

[108] **Thompson, F. L., Li, Y., Gomez–Gil, B., Thompson, C. C., Hoste, B., Vandemeulebroucke, K., Rupp, G. S., Pereira, A., De Bem, M. M., Sorgeloos, P. & Swings, J. (2003).** *Vibrio neptunius* sp. nov., *V. brasiliensis* sp. nov. and *V. xuii* sp. nov., isolated from the marine aquaculture environment (bivalves, fish, rotifers and shrimps). *Int J Syst Evol Microbiol* **53**, 245–252.

[109] **Thompson, F. L., Iida, T. & Swings, J. (in press).** Biodiversity of vibrios. *Microbiol Mol Biol Rev*.

[110] **Thompson, F. L., Gevers, D., Dawyndt, P., Thompson, C. C., Naser, S., Hoste, B., Munn, C. & Swings J. (submitted).** Identification of vibrios using rpoA gene sequences.

[111] **Tibshirani, R., Walther, G. & Hastie, T. (2000).** Estimating the number of clusters in a dataset via the gap statistic. Tech. Rep. **208**, Department of Statistics, Stanford University.

[112] **Ukkonen, E. (1985).** Algorithms for approximate string matching. *Information and Control* **64**, 100–118.

[113] **Urakawa, H., Kita-Tsukamoto, K. & Ohwada, K. (1999).** Reassessment of the taxonomic position of *Vibrio iliopiscarius* (Onarheim et al. 1994) and proposal for *Photobacterium iliopiscarium* comb. nov. *Int J Syst Bacteriol* **49(1)**, 257–260.

[114] **Vandenberghe, J., Verdonck, L., Robles–Arozarena, R., Rivera, G., Bolland, A., Balladares, M., Gomez–Gil, B., Calderon, J., Sorgeloos, P. & Swings, J. (1999).** *Vibrios* associated with *Litopenaeus vannamei* larvae, postlarvae, broodstock and hatchery probionts. *Appl Environ Microbiol* **65**, 2592–2597.

[115] **Vauterin, L. & Vauterin, P. (1992).** Computer-aided objective comparison of electrophoresis patterns for grouping and identification of microorganisms. *Eur Microbiol* **37**, 37–41.

[116] *Vibrio fischeri* genome project.
`http://ergo.integratedgenomics.com/Genomes/VFI/`.

[117] **Vezzi, A., Campanaro, S., D'Angelo, M., Simonato, F., Vitulo, N., Lauro, F., Cestaro, A., Malacrida, G., Simionati, B., Cannata, N., Bartlett, D. & Valle, G. (submitted).** Genome Analysis of *Photobacterium profundum* reveals the complexity of high pressure adaptations. Accession Number: CR354532.

[118] **Vos, P., Hogers, R., Bleeker, M., Reijans, M., van de Lee, T., Hornes, M., Frijters, A., Pot, J., Peleman, J., Kuiper, M. & Zabeau M. (1995).** AFLP: a new technique for DNA fingerprinting. *Nucleic Acids Res* **23(21)**, 4407–4414.

[119] **Ward, J. H. (1963).** Hierarchical grouping to optimize an objective function. *J Am Stat Assoc* **58**, 236–244.

[120] **Waterman, M. (1995).** Introduction to Computational Biology. Chapman & Hall, London, UK.

[121] **Wedel, M. & Kamakura, W. A. (1998).** Marketing Segmentation. Conceptual and Methodological Foundations. In: Mixture Models, 89–92. Kluwer Academic, Boston Dordrecht London.

[122] **Wayne, L. G., Brenner, D. J., Colwell, P. R., Grimont, P. A. D., Kandler, O., Krichevsky, M. I., Moore, L. H., Moore, W. E. C., Murray, R. G. E., Stackebrandt, E., Starr, M. P., Trüper, H. G. (1987).** Report of the ad hoc committee on reconciliation of approaches to bacterial systematics. *Int J Syst Bacteriol* **37**, 463–464.

[123] **Willems, A., Doignon-Bourcier, F., Coopman, R., Hoste, B., de Lajudie, P. &Gillis M. (2000).** AFLP fingerprint analysis of *Bradyrhizobium* strains isolated from *Faidherbia albida* and *Aeschynomene* species. *Syst Appl Microbiol* **23(1)**, 137–147.

Chapter 5

Improving the Discriminatory Power of Bacterial Whole Cell Fatty Acid Methyl Ester Analysis

"In theory, there is no difference between theory and practice. In practice, there is."

— Chuck Reid

THE current chapter examines how the discriminatory power of a bacterial whole cell fatty acid identification system can be significantly enhanced, by exploring the vast amounts of information accumulated during fifteen years of routine gas chromatographic analysis on the fatty acid content of environmental aerobic bacteria. This large knowledge base is established as a collaborative effort between the Laboratory of Microbiology at the Ghent University and the BCCM™/LMG Bacteria Collection. Construction of a global peak occurrence histogram is shown to serve as a highly informative tool for assessing the delineation of naming windows, used during the automatic recognition of fatty acid compounds. Along the lines of this data mining application, it is suggested that several naming windows of the Sherlock MIS TSBA50 peak naming method may need to be re-evaluated in order to fit more closely with the bulk of observed fatty acid profiles. Simultaneously, the peak occurrence histogram instigated the delineation of 32 new peak naming windows, accounting for a 26% increase in the total set of fatty acid features taken into consideration for bacterial identification. By scrutinizing the relationships between the newly delineated naming windows and the many taxonomic units covered within the proprietary fatty acid database, all new naming windows were proven to correspond with stable features of some specific groups of microorganisms. This latter analysis clearly underscores the impact of incorporating the new fatty acid compounds for improving the resolution of the bacterial identification system and endorses the applicability of knowledge discovery in databases within the field of microbiology.

5.1 Introduction

Variations in the fatty acid content of bacterial cells have been widely used for bacterial classification and identification for more than 40 years. Ever since the introduction of gas chromatographic analysis of cellular fatty acids by Abel *et al.* [1], this technique has been frequently applied in various taxonomic studies [49]. Numerous scientific papers have used the fatty acids between 9 and 20 carbons in length to characterize genera and species of bacteria, especially nonfermentative Gram negative microorganisms [34]. With the many improvements in automated calibration and interpretation of the chromatographic profiles, reproducible fatty acid profiles nowadays can be generated rapidly, provided that strains are grown under specified standardized conditions [25], and the identification of microorganisms by analysis of their cellular fatty acid composition has become a routine method in many laboratories.

The construction and the information content of a large data warehouse that contains the results of a long-term gas chromatographic analysis study on the fatty acid content of a broad diversity of environmental aerobic bacteria, established as a collaborative effort between the Laboratory of Microbiology at the Ghent University and the BCCM™/LMG Bacteria Collection, is outlined in section 5.2. After reviewing the occurrence of fatty acids in the bacterial cell (most of them are found within the cytoplasmic membrane as constituents of the polar lipids and glycolipids [41]), we explain how the Sherlock Microbial Identification System (MIS; Microbial ID, Inc. (MIDI), Newark, Delaware, USA) was employed for the separation and identification of cellular fatty acid methyl esters (FAME). This software package was chosen merely because of its library generation capability, the large number of known compounds recognized by the system's peak naming tables, the ability to compare a large number of strains over a period of time, and the routine use of this system for bacterial analysis in many laboratories [38]. The chromatographic fatty acid peaks are automatically named and quantified by the system, and the wealth of information contained in these compounds can be used for bacterial identification by considering not only the presence or absence of each acid, but also by using the data in quantitative fashion [34]. In order to fully exploit all opportunities offered by the assemblage of bacterial fatty acid content information, the data is routinely transfered to a relational database system that is accessible through the ODBC interface of the BioNumerics software package (Applied Maths, Sint-Martens-Latem, Belgium), which offers an improved set of tools for data management and online transactional processing (OLTP). Finally, it is explained how the data was further predigested in terms of the data warehousing paradigm, as to improve the overall performance for more complex data mining applications [17, 20].

Some properties about the distribution of the bacterial whole cell fatty acid content are unraveled in section 5.3, by only taking into account the presence and absence of the fatty acid compounds recorded into the proprietary FAME database for a broad variety of microorganisms. A review is given on the frequency of occurrence for all peaks named by means of the Sherlock MIS TSBA50 method and it is demonstrated how the information in the FAME database can be employed to test hypotheses concerning the uniqueness of qualitative fatty acid templates for a certain group of microorganisms. Additionally, it is explored how the construction of a global peak occurrence histogram might contribute to a

better delineation of the peak naming windows, applied for the automatic identification of fatty acid compounds on the basis of their location in the chromatograms. In this context, several peaks were detected in the peak occurrence histogram, which do not correspond with one of the existing naming windows currently defined in the Sherlock MIS TSBA50 peak naming table. In order to check whether these yet unnamed histogram peaks correspond with stable fatty acid compounds, section 5.4 attempts to predict the significance of each peak for the different taxonomic units present in the database, taking into account that the discovery of new fatty acid compounds may affect the estimated total quantity of fatty acids for some strains, and also the overall relationships between strains as observed from fatty acid analysis [38]. Finally, pairwise database identification is proposed as an alternative identification technique opposed to the library identification approach implemented in the Sherlock MIS.

5.2 FAME database construction

5.2.1 Cellular fatty acids

Despite their differences, most cells of the living organisms dwelling our blue planet have a great deal in common with each other. Every cell, whether an archebacterium living in a superheated sulphur vent at the bottom of the ocean or a cell in a hair follicle within the fur of a two-ton polar bear roaming the arctic circle span has certain basic qualities: they contain *cytoplasm* and *genetic material*, are enclosed in a *membrane* and have similar basic mechanisms for translating genetic messages into the main type of biological molecule, the *protein*. Membranes are the boundaries between the cell and the outside world. All present day cells have a *phospholipid* cell membrane. Phosopholipids are lipids (oils or fats) with a phosphate group attached. The end with the phosphate group is *hydrophilic* (attracted to water) and the lipid is *hydrophobic* (repelled by water). Cell membranes generally consist of two layers of these molecules, with the hydrophobic ends facing in, and the hydrophilic ends facing out. This keeps water and other materials from getting through the membrane, except in a controlled way through special pores and channels. A lot of the action in cells happens at the membrane. For single celled organisms such as bacteria, the membrane contains molecules that can sense the environment. Some bacterial cell walls can surround and engulf food, or attach and detach parts of themselves in order to move around. The bacterial cell membrane also plays a crucial role in the energy production, by maintaining a large acidity difference between the inside and the outside of the cell [12].

Chemotaxonomy is based on investigating the chemical composition of bacterial cells, such as their cellular fatty acids, mycolic acids, polar lipids, quinones, polyamines, cell wall compounds and exopolysaccharides [43]. The *fatty acids* can be defined as carboxylic acid derivatives of long-chain aliphatic molecules, such as lipids and lipopolysaccharides. Just as all the other organic molecules they are expressions of the information encoded in the nucleic acids (DNA), so they can be used as taxonomic markers provided they are stable and discontinuously distributed. In bacteria, fatty acids range in chain length from

2 carbon molecules to over 90 carbon components, as found in mycolic acids. Taxonomically, fatty acids within a range between 10 and 24 carbon molecules provide the most important source of information and are present across a diverse range of microorganisms. Lower molecular weight compounds are usually associated with metabolism, e.g. fermentation end-products, and are distinct from cellular or structural fatty acids. Although they might also contain useful taxonomic information, they are usually not considered for the characterisation of bacterial strains [37].

A wide variety of lipids are present in bacterial cells, but most fatty acids are found within the cytoplasmic membrane as constituents of polar and glycolipids, where they form an integral part of the lipid bilayer (Figure 5.1). Other types of lipids, such as sphingophospholipids, only occur in a restricted number of taxa and were shown to be valuable discriminatory features within these groups [18]. Structural fatty acids are also present in the outer membrane of gram-negative bacteria as constituents of the lipopolysaccharide (Figure 5.2). Consequently, whole-organism hydrolysates of gram-negative bacteria not only contain fatty acids from the cytoplasmic membrane, but also those from the outer membrane. In lipopolysaccharide, the 3-hydroxy fatty acids form a characteristic component of Lipid A and their ubiquitous but discontinuous distribution has made them valuable taxonomically. The diversity in fatty acid types (chain lengths, double bond positions and substituent groups) and their highly regulated production makes them useful taxonomic markers. Mostly, the total cellular fatty acid fraction is extracted, but particular fractions such as polar lipids have also been analyzed separately [7, 41].

More than 300 different chemical structures of fatty acids have been identified. The wealth of information contained in these compounds can be estimated by considering not only the presence or absence of each acid, but also by using the data in a quantitative fashion. While the theoretical ability to differentiate amongst 2^{300} different combinations is not practical due to the nonrandom distribution within groups of bacteria, the huge number of fatty acids creates some descriptive opportunities for defining bacterial taxa [34]. An overview of the various kinds of fatty acids found in bacteria is assembled in Figure 5.3. Fatty acids can be classified according to the basic structure of their carbon skeleton, i.e. the number of carbon atoms, the number and position of the double bonds in the carbon chain, and the presence of functional groups. The chemical composition of the straight chain palmitic acid, written as 16:0, is shown in Figure (*i*). The number before the colon refers to the number of carbons in the compound, the number after the colon indicates the number of double bonds in the carbon chain. The *carboxyl group* (COOH) is at the right. In Figure (*ii*), the designation 16:1 indicates that the compound has 16 carbons and 1 double bond. The chemical structure represents the unsaturated fatty acid 16:1 ω7c. Note that both hydrogens at the double bond are on the same side in the *cis* conformation. The ω7c notation refers to the 7th carbon from the ω-end of the chain. When counting from the carboxyl group that is located at the reverse α-end of the chain, the same compound can be equivalently noted as 16:1 *cis* 9. Fatty acids with unknown double bond positions are differentiated using capital letters, as in 15:1 iso F and 15:1 anteiso A. Figure (*iii*) shows the unsaturated fatty acid 16:1 ω7t. This compound is in *trans* conformation, because the hydrogens at the double bond are on opposite sides of the molecule. Biosynthetically, fatty acids with odd carbon numbers can be considered as a different series of acids from those with even carbon numbers. In

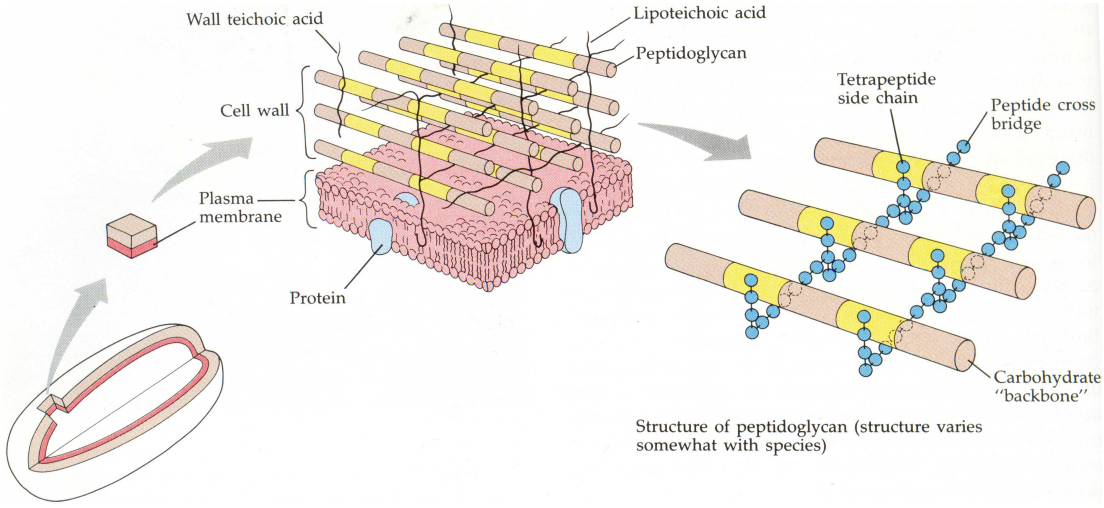


Figure 5.1: Bacterial cell wall of the gram-positive bacteria, showing the structure of peptidoglycan. Together, the carbonhydrate backbone (glycan portion) and amino acids (peptide portion) make up peptidoglycan. The frequency of peptide cross bridges and the number of amino acids in these bridges vary with the bacterial species (picture taken from Tortora *et al.* [39]).

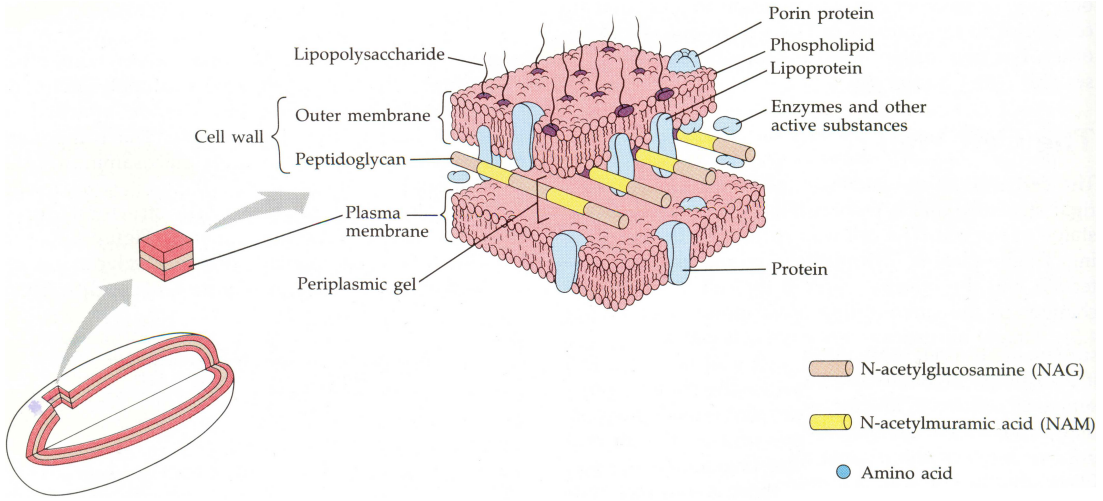


Figure 5.2: Bacterial cell wall of the gram-negative bacteria (picture taken from Tortora *et al.* [39]).

general, unsaturated fatty acids (those containing double bonds) of bacteria are monoenoic or monounsaturated, that is, they have only one double bond. In contrast, polyenoic fatty acids have two or more double bonds and a more limited distribution. The position of the double bonds is biosynthetically significant and hence of taxonomic value.

Fatty acids can be subdivided into two groups, namely straight chain and branched fatty acids. The latter include iso-, anteiso-, 10-methyl- and dimethyl acetal-branched fatty acids, and also the cyclopropane fatty acids. Iso- and anteiso-branched fatty acids are methyl-branched fatty acids at the second and third carbon from the ω-end (non-carboxyl end) of the carbon chain, respectively, and are unique to bacteria. Examples are shown in Figures (*iv*) and (*v*), where the *methyl group* (CH_3) occurs at the second, respectively third, to the last carbon in the chain. Similarily, the 10-methyl-branched fatty acids, such as 17:0 10-methyl depicted in Figure (*vi*) have a methyl group at the tenth carbon position counted from the α-end. 18:0 10-methyl is also known as tuberculostearic acid (TBSA) or 10-methyloctadecanoic acid. Figure (*vii*) represents the cyclopropane fatty acid 17:0 cyclo ω7c. Again, when counting from the α-end of the chain, the same compound can be alternatively noted as 17:0 cyclo 9-10. This molecule is composed of 16:1 ω7c, with addition of an extra carbon group at the double bond position. Dimethyl acetal 16:0, shown in Figure (*viii*), is abbreviated to 16:0 DMA. Dimethyl acetals occur as analogs of the fatty acids present in anaerobic bacteria, and can contain any of the functional groups present in the fatty acids of (*iv-vii*).

The normal hydrocarbon of 16 carbon atoms in length, written as n 16:0, is depicted in Figure (*ix*), while Figure (*x*) represents aldehyde 16:0. Examples of 2-hydroxy and 3-hydroxy fatty acid molecules with 16 carbons are shown in Figures (*xii*) and (*xiii*), having a hydroxyl group added at the 2nd (α) and 3rd (β) position respectively. Hydroxyl groups may occur at other positions besides the second and third carbons. However, these are rarely found among bacterial species. Combinations of the various functional groups discussed above also occur. As an illustration of mixing different structural elements, we refer to the fatty acid 17:0 ISO 2OH that is given in Figure (*xiv*), combining an iso-branched methyl group with a hydroxyl group at the second carbon from the ω-end in the carbon chain. Finally, the chemical structure in Figure (*xv*) represents the fatty acid methyl ester 16:0, where a methyl group is added to the carboxyl group on the right to increase the volatility of the fatty acids, althus enabling gas chromatographic (GC) separation of the cellular fatty acids. This compound is noted as 16:0 FAME.

5.2.2 Chromatographic fatty acid decomposition

Bacterial fatty acids, unlike many other phenotypic characteristics, are genetically highly conserved, owing their essential role in cell structure and function. In addition, technical advances through the development of fused-silica capillary columns, automatic injection systems, digital integrators and standardized calibrators have increased the applicability of whole-cell fatty acid analysis by gas-liquid chromatography [24]. The most stable and reproducible cellular fatty acid profiles are achieved by carefully regulating the growth con-

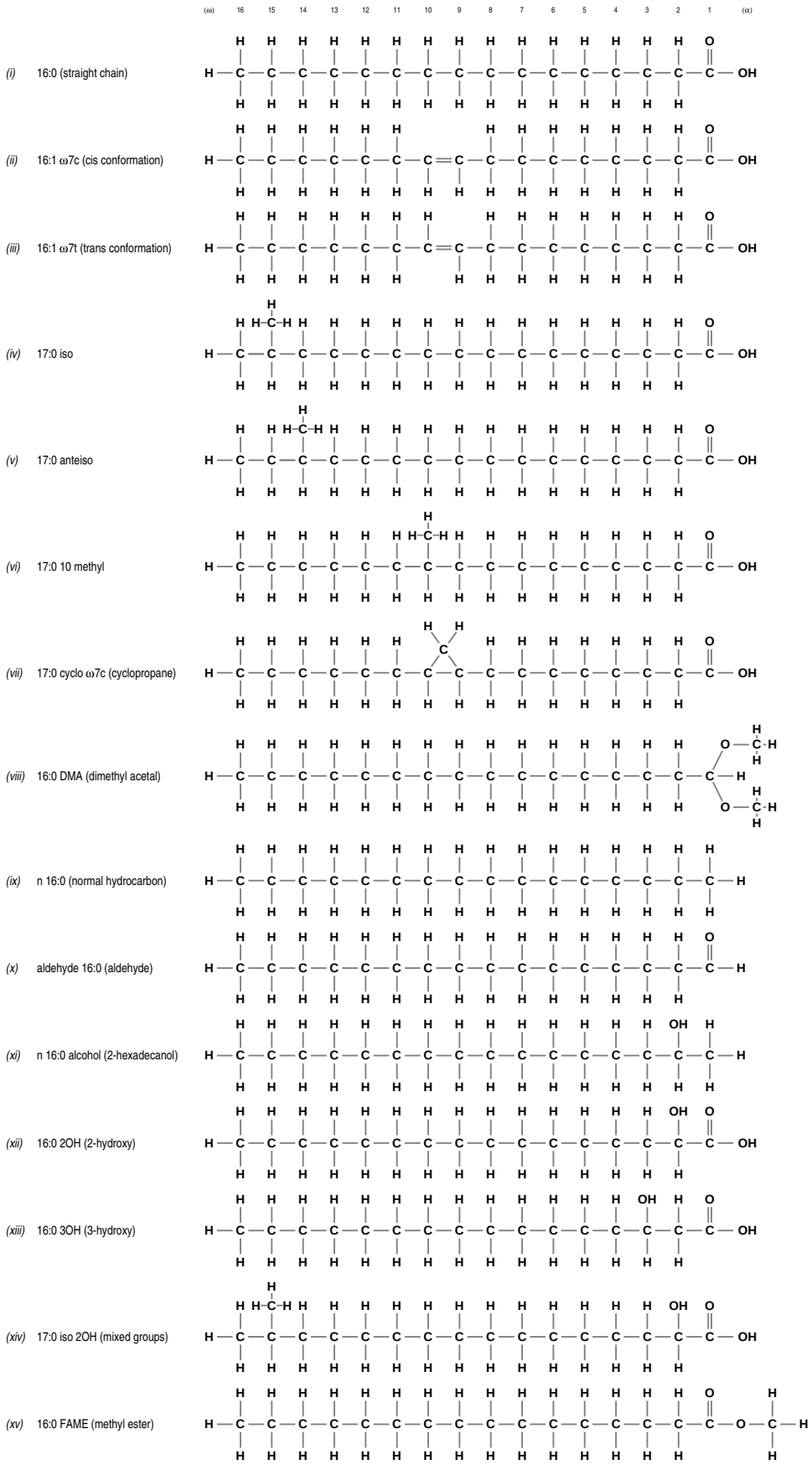


Figure 5.3: Nomenclature of fatty acids

ditions [34]. After all, several scientific papers have reported on the effects of physiological age of the cell cultures, growth temperature, incubation time, and cultivation medium on the bacterial fatty acid composition [16]. However, differences in growth conditions and extraction procedures influence only the quantitative content of the fatty acid methyl esters in most cases, rather than the overall qualitative image of the fatty acid composition [41]. To minimize these variables, a standard protocol was accurately followed as much as possible during the construction of an in-house bacterial fatty acid composition database covering a broad spectrum of environmental aerobic microorganisms. Such a standardized protocol must be carefuly devised, to accomodate good growth for the majority of bacteria, and the protocol we have followed was especially designed for the construction of the TSBA50 identification library that is commercially exploited within the Sherlock Microbial Identification System (MIS; Microbial ID, Inc. (MIDI), Newark, Delaware, USA). Most aerobic bacteria will grow well on the prescribed Trypticase Soy Broth Agar (TSBA), which consists of 30 gl^{-1} Trypticase Soy Broth (BBL) and 15 gl^{-1} of Bacto Agar (Difco). The chosen growth period is 24 hours at a fixed temperature of 28°C, and the effect of physiological age is minimized in the broth cultures by harvesting cells from a streak on the overlap between the second and third quadrant on the plate. Those bacteria that will not grow under the prescribed conditions are cultivated according to the conditions which would be most commonly used for their growth in the laboratory. These deviating conditions are carefully noted down into the fatty acid database, in order to enable a correct interpretation of the fatty acid profiles during later stages of data analysis.

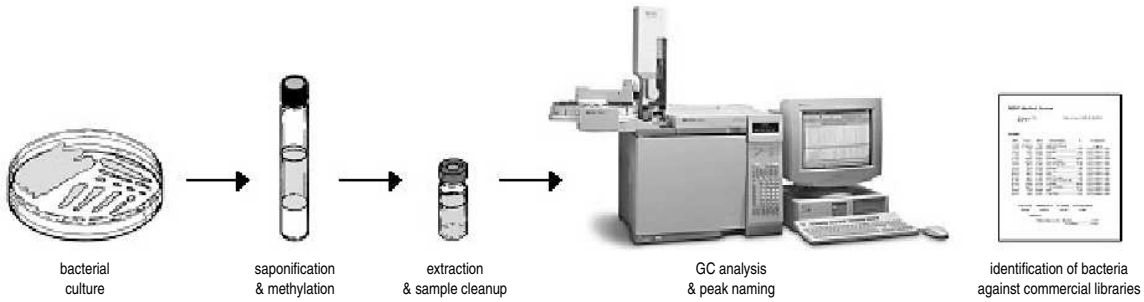
The method applied for the extraction and derivation of bacterial whole-cell fatty acid methyl esters was initially described by Miller [29]. Briefly, it takes the following five steps to prepare the GC ready extracts. Approximately 40 to 50 mg (wet weight) of bacterial cells is *harvested* from the streaked plate, and placed into a clean test tube (13×100 mm) with a Teflon-lined screw cap. Cells are then *saponified* by heating them at 100°C for 30 min after the addition of 1.0 ml of 15% NaOH (w/v) in 50% aqueous methanol (v/v). The hydrolysate is then cooled to ambient temperature, 3.25 N of HCl in 45.8% methanol is added, and the mixture is heated at 80°C for 10 min (this step is critical in time and temperature). This drops the pH of the solution below 1.5 and causes *methylation* (for the increased volatility in a partially polar column) of the fatty acid. The fatty acid methyl ester is poorly soluble in the aqueous phase at this point. The methylated fatty acids are then quickly cooled down to ambient temperature and *extracted* through the addition of 1.25 ml of hexane/methyl tert-butyl ether (1:1 vol/vol), after which the tubes are capped and gently shaken for about 10 minutes. This will extract the fatty acid methyl esters into the organic phase for use with the gas chromatograph. Subsequently, the tubes are uncapped and the aqueous (lower) phase is pipetted out and discarded. Finally, in order to reduce contamination of the injection port liner, the column and the detector, the sample is *washed* by adding 1.2% of dilute NaOH (w/v) to the remaining organic layer. The base washing removes underivatized fatty acids and trace amounts of HCl from the solution, which degrade the column and distort the peak shape of hydroxy fatty acids in subsequent runs [22]. Approximately two-thirds of the organic layer containing the fatty acid methyl esters (FAMEs) is then transferred to a septum-capped sample vial for GC analysis. The previous procedure is summarized on top of Figure 5.4.

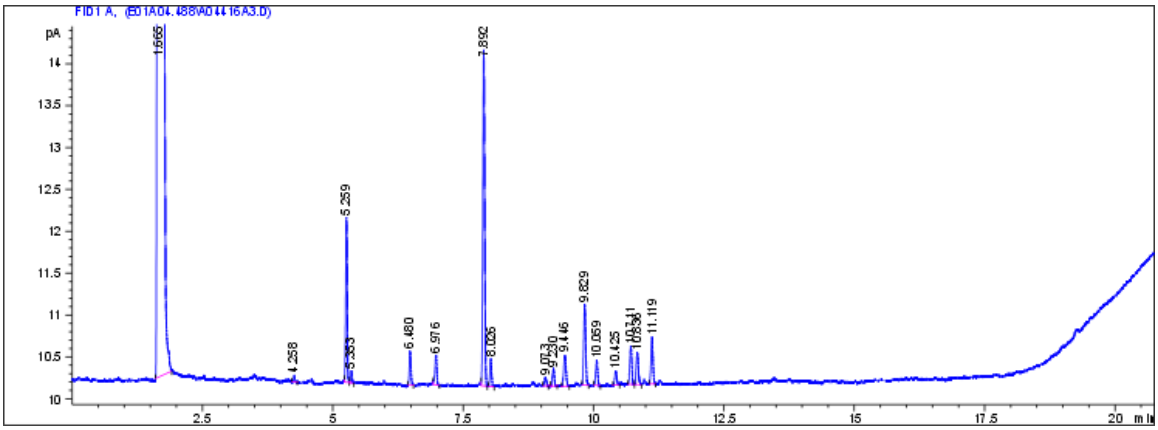
After preparation, fatty acid methyl esters were analyzed on a HP 6890A gas chromatograph (Hewlett-Packard Co., Avondale, Pennsylvania, USA) equipped with a flame ionization detector, automatic sampler and computer. Gas-liquid chromatographic separation of the fatty acid methyl esters was achieved with a fused-silica capillary column (25 m $\times$ 0.2 mm) coated with cross-linked 5% phenylmethyl silicone (film thickness 0.33 μm; HP Ultra 2). The specific operating parameters of the instrument are controlled and set automatically by the ChemStation software (version 4.02, Hewlett-Packard). This software package is tightly coupled to the Sherlock MIS and is used for operating sampling, analysis, and integration of the chromatographic samples. The user specified parameters are as follows: injector temperature, 250°C; detector temperature, 300°C; and oven (column) temperature, programmed from 170°C to 300°C at 5°C/min and held at 300°C for 5 min prior to recycling. The flame ionization detector allows for a large dynamic range and provides good sensitivity. Hydrogen is the carrier gas, nitrogen is the 'make-up' gas, and air is used to support the flame. Until the year 2000, the chromatograms with peak retention times and areas were produced on a recording integrator coupled to a HP 5890A gas chromatograph and were electronically transferred to the computer for analysis, storage and chromatographic report generation. After switching to a HP 6890A gas chromatograph, the electronic signals from the GC detector were directly passed over to the computer where the task of peak recognition and area determination was taken over by the ChemStation software, but the results generated before and after that technical alteration remained mutually comparable. A typical chromatogram generated for the *Bacillus cereus* type strain LMG 6923^{T} is shown in the middle of Figure 5.4.

5.2.3 Calibration and cellular fatty acid identification

Even after accurately following well-established incubation protocols and GC operating conditions, results can be distorted by ordinary equipment drift and inter-lab environmental differences. To achieve reproducibility, Sherlock MIS regularly calibrates the chromatographic unit using an external calibration standard developed and manufactured by MIDI. The standard is a mixture of the straight chain saturated fatty acids from 9 to 20 carbons in length (9:0 to 20:0) and five hydroxy acids. All compounds are added quantitatively so that the gas chromatographic performance may be routinely evaluated by the MIDI software each time the calibration mixture is analyzed. The hydroxy compounds are sensitive to small changes in pressure/temperature relationships and to contamination of the injection port liner. As a result, these calibration compounds also function as quality control checks for the system [34]. A sample containing a fresh calibration mixture is processed several times at the start of every new batch run of bacterial samples, and is rerun after every user-defined number of samples to allow for recalibration during batch processing.

Retention time data obtained from injecting the calibration mixture is converted into *equivalent chain length* (ECL) units for bacterial fatty acid naming. The ECL value for each fatty acid peak can be derived as a linear interpolation of its elution time in relation to





Sherlock Version: 3.10 DATA7:E01A04488A 31-MAR-04 17:52:42

ID: 4416 BACI-CEREU (LMG 6923 T/QC9/01/KW3/P20) Date of run: 04-OCT-01 23:12:07
Bottle: 19 SAMPLE [TSBA40]

RT	Area	Ar/Ht	Respon	ECL	Name	%	Comment 1	Comment 2
1.665	440858438	0.028	. . .	6.992	SOLVENT PEAK	. . .	< min rt	
4.258	269	0.029	1.027	11.609	12:0 ISO	0.56	ECL deviates -0.000	Reference -0.002
5.259	7723	0.031	0.995	12.614	13:0 ISO	15.48	ECL deviates 0.000	Reference -0.002
5.353	528	0.029	0.993	12.701	13:0 ANTEISO	1.06	ECL deviates -0.001	Reference -0.003
6.480	1814	0.034	0.975	13.619	14:0 ISO	3.56	ECL deviates 0.000	Reference -0.002
6.976	1702	0.039	0.970	14.002	14:0	3.32	ECL deviates 0.002	Reference -0.001
7.892	18967	0.037	0.964	14.623	15:0 ISO	36.83	ECL deviates 0.000	Reference -0.002
8.026	1497	0.037	0.963	14.714	15:0 ANTEISO	2.90	ECL deviates 0.001	Reference -0.002
9.073	505	0.036	0.960	15.389	16:1 w7c alcohol . .	0.98	ECL deviates 0.002	
9.230	1016	0.041	0.960	15.487	Sum In Feature 2 . .	1.97	ECL deviates -0.001	14:0 3OH/16:1 ISO I
9.446	2257	0.049	0.960	15.621	16:0 ISO	4.36	ECL deviates -0.006	Reference -0.009
9.829	5040	0.042	0.959	15.859	Sum In Feature 3 . .	9.74	ECL deviates 0.007	15:0 ISO 2OH/16:1w7c
10.059	1512	0.037	0.959	16.002	16:0	2.92	ECL deviates 0.002	Reference -0.000
10.425	849	0.040	0.959	16.220	15:0 2OH	1.64	ECL deviates 0.001	
10.711	2401	0.041	0.959	16.390	ISO 17:1 w10c . . .	4.64	ECL deviates 0.002	
10.836	2233	0.046	0.959	16.463	ISO 17:1 w5c	4.32	ECL deviates 0.002	
11.119	2960	0.041	0.959	16.631	17:0 ISO	5.72	ECL deviates 0.001	Reference -0.002
*******	1016	. . . . .		. . .	SUMMED FEATURE 2 . .	1.97	12:0 ALDE ?	unknown 10.928
*******	. . . . .	. . . . .		. . .	. . . . .	. . .	16:1 ISO I/14:0 3OH	14:0 3OH/16:1 ISO I
*******	5040	. . . . .		. . .	SUMMED FEATURE 3 . .	9.74	16:1 w7c/15 iso 2OH	15:0 ISO 2OH/16:1w7c

Solvent Ar	Total Area	Named Area	% Named	Total Amnt	Nbr Ref	ECL Deviation	Ref ECL Shift
440858438	51273	51273	100.00	49634	10	0.003	0.003

TSBA50 [Rev 5.0] Bacillus 0.602
B. cereus 0.602
B. c. GC subgroup A 0.602
B. thuringiensis 0.509
B. t. canadensis 0.509
B. t. israelensis 0.416
B. t. kurstakii 0.372

Figure 5.4: Fatty acid methyl ester extraction procedure (top), sample chromatographic report (middle) and sample composition report (bottom) resulting from automated fatty acid separation by the Sherlock Microbial Identification System.

the elution times of the known series of straight chain fatty acids of the calibration mixture

$$\mathrm{ECL}_x = n + \frac{\mathrm{RT}_x - \mathrm{RT}_n}{\mathrm{RT}_{n+1} - \mathrm{RT}_n}\,, \tag{5.1}$$

where RT_x is the retention time of the unknown fatty acid x. RT_n represents the retention time of the saturated fatty acid methyl ester (having n carbon atoms) preceding x in the calibration mixture, and RT_{n+1} is the retention time of the saturated fatty acid methyl ester eluting after x in the calibration mixture [34]. As such, the saturated fatty acids of the calibration mixture are assigned an ECL value corresponding to their chain length (e.g. 11:0 $\equiv$ ECL 11.000). Calibration thus accounts for large changes in absolute retention times as long as the relative positions remain unchanged. In this respect, calibration of fatty acid profiles is performed in complete analogy with the normalization of gel electrophoresis patterns as discussed in Chapter 4, where the external calibration mixture takes on the role of an external molecular weight marker.

Thus, it is possible by interpolation with the external calibration standard to compute the ECL value for each detected compound following a GC analysis. The cellular fatty acids of the unknown samples (i.e. the non-calibration samples) are then identified by matching their ECL values with the naming windows of a predefined peak naming method. In essence, a peak naming table is composed of a series of ECL windows of variable width (most naming windows are between 0.020 and 0.030 ECL units wide) and a name for the fatty acid compound that corresponds with most or all chromatographic peaks that fall within each particular window. The exact structure of the compound that corresponds with a peak in a chromatogram is generally determined using mass spectrometry. But once the relation between the chemical structure and the chromatographic position is known, the naming window approach avoids the extra step of mass spectroscopic analysis for the identification of the fatty acids. An example of the TSBA50 (version 5.0) peak naming method that is used by the Sherlock MIS for naming the fatty acids of aerobic bacteria grown on TSBA medium is shown in section B.3. Apart from the commercially available peak naming methods, Sherlock MIS also contains the necessary tools that support the creation of user-defined peak naming tables. After naming the peaks in an unknown sample, Sherlock MIS compares the ECL values for most stable series (e.g. saturated straight chain or branched chain acids) to the peak naming table's theoretically perfect values and may recalibrate internally if sufficient differences are detected. This feature allows the system to be up and running for two days unattended without worrying about drift between runs [34].

Practical constraints like the length of the capillary column and the limited run time force acceptance of less than perfect chromatography. As a result, some peaks will not be clearly separated and their corresponding naming windows in the peak naming table are overlapping. Because it is essential for further computational analysis not to separate equivalent fatty acids during the peak naming process, the Sherlock approach makes use of so-called *summed features* wherever imperfect peak discrimination occurs. This means that although normalized peaks are still associated to the window having its center closest to the observed peak (which carries the correct name for the compound in the majority of cases), the overlapping windows are regarded as a single character when comparing vectors of named features. When multiple gas chromotographic peaks are detected for the composing windows of a summed feature, their peak areas are added up. This extinguishes harmful

effects of incorrect peak identification during cluster analysis and pattern recognition. Several examples of predefined summed features can be found in the TSBA50 peak naming method outlined in section B.3, where clusters of associated naming windows are indicated by equal negative identifiers. For this peak naming table, summed feature 1 comprises any combination of 15:1 iso H, 15:1 iso I and 13:0 3OH. Summed feature 2 is composed of 12:0 aldehyde, unknown 10.928, 16:1 iso I and 14:0 3OH. Summed feature 3 comprises 15:0 iso 2OH, 16:1 ω7c, or both. Summed feature 4 collects 17:1 iso I, 17:1 anteiso B, or both. 18:2 ω6,9c, 18:0 anteiso, or both together form summed feature 5. Summed feature 6 comprises 19:1 ω11c, 19:1 ω9c, or both. Summed feature 7 comprises any combination of unknown 18.846, 19:1 ω6c and 19:0 cyclo ω10c. As can be observed from the definition of summed feature 2, the summed feature approach can also be employed for the agglomeration of non-overlapping peak windows that correspond with closely related compounds. In this particular case, the 12:0 aldehyde compound is a breakdown product of the 14:0 3OH fatty acid. The breakdown becomes more and more significant as the injection port liner gets older and dirtier. The relative amounts of these fatty acids vary over the lifetime of the liner, but the sum of the peaks remains stable [22]. Note also that the chemical composition of some chromatographic peaks has not as yet been determined by mass spectroscopy. Unknown fatty acids are therefore designated with the term *unknown* followed by the equivalent chain length of their naming window center, so that they can be incorporated in computation analysis of the fatty acid profiles.

Following computer analysis by the Sherlock MIS, a fatty acid identification report is generated for any sample run, which contains a variety of parameters concerning each peak in the chromatogram. These include retention time, area, area/height ratio, response factor, equivalent chain length, peak name (specific fatty acid), and relative amount of the fatty acid present in the cell, as well as some calibration information. It is the expression of the peak area values as percentage of the total area of all named peaks in the chromatogram (exclusive of the solvent front) that will be used for comparison of different fatty acid profiles. Because the area measured from the chromatogram increases towards the end of a profile, peak areas in the beginning of a profile are somewhat underestimated, whereas peak areas at the end of a profiles are slightly overestimated. Peaks in the early part of the analysis are more affected by GC oven temperatures and those later in the analysis are more severely impacted by carrier gas flow rates. The use of an electronic pressure controller to achieve constant flow minimizes the latter type of error in the gas chromatograph [34]. In order to attain an objective approximation of the relative fatty acid amount a_i^r of the ith named peak, a weighted expression is used

$$a_i^r = \frac{r_i \, a_i^a}{\sum_{j \in \mathcal{N}_p} r_j \, a_j^a} \,, \tag{5.2}$$

where $\mathcal{N}_p$ represents the set of named peaks of the profile, a_k^a is the absolute area of the kth peak, and r_k is the weight factor assigned according to the ECL position of the kth peak. In the Sherlock MIS jargon, these weights are termed *response factors*, and their value is calculated by interpolation from a comparison between the known and measured concentrations of the compounds in the calibration mixture. Remark that the sum of the relative fatty acid amounts for a given profile, as defined in (5.2), is 1.

An example report derived from the chromatogram of the *Bacillus cereus* type strain LMG 6923^T is shown at the bottom of Figure 5.4. For the fatty acid peak 13:0 anteiso, the measured retention time was 5.353 min. The integrated area under the peak amounts to 528, with an area/height ratio of 0.029. This latter value is important for quality control of the samples, because during the analysis of materials such as fatty acids, the extraction procedure may carry over sterols, non-methylated fatty acids and other non-fatty acid materials. Additionally, electronic noise may result in transient spikes, which might interfere with the chromatography. Fatty acid peaks always have area/height ratios greater than 0.017 and less than 0.070, making it possible to ignore peaks if values are found outside of this interval. Electronic noise spikes are typically less than 0.017, whereas non-fatty acid methyl ester peaks (carryover, sterols, etc.) are usually greater than 0.070 [34]. After normalization with the calibration mixtures run before the profile, an equivalent chain length of 12.701 is found, and the peak can be named by finding its associated window in the peak naming table. The total area of the named peaks in the fatty profile amounts to 51273, but according to the denominator of (5.2) this value should be adjusted to 49634 (included in the sample report as the *total amount*). As a result, the relative contribution of this peak to the total area is $a_r = \frac{528 \times 0.993}{49634} \times 100 = 1.06\%$. The first comment for this peak in de fatty acid composition report indicates that this peak has emerged faster than expected by one thousandth of an ECL unit, when compared to the center of the window in the TSBA peak naming table.

5.2.4 Library identification of bacteria

In this subsection we explain how the Sherlock MIS performs identification of bacterial strains with an unknown taxonomic position, from the knowledge of their cellular fatty acid composition. Identification analysis of the fatty acid profile of an unknown sample happens by comparison to the entries of a predetermined identification library. For the construction of the commercial identification libraries that are optionally delivered with the distribution of the Sherlock MIS, several ten thousands of fatty acid patterns from well-characterized type and reference strains were generated according to a strict protocol of sample growth and fatty acid extraction procedures, in order to avoid major qualitative and quantitative variability in the fatty acid profiles caused by these influencing factors. To attain comparability with a particular identification library, the whole-cell fatty acid methyl ester composition for the unknown microorganism should be determined according to the same incubation protocol as used for the samples during library generation. Apart from the commercially available identification libraries, the Sherlock Library Generation Software package enables the creation of custom identification libraries that are trained by a selected set of proprietary microorganisms. Again, a well-described protocol of growth conditions should be adhered to the construction of such libraries.

Each entry from an identification library must be composed of the fatty acid profiles generated for some type and reference cultures that form a coherent taxonomic unit, e.g. a species or a subspecies. For the commercial libraries that are provided with the Sherlock MIS, strains were collected from experts and culture collections around the world, in order

to avoid a potential geographical bias. Wherever possible, 20 or more strains of a species or subspecies were analyzed for the creation of the library entry, as to gain some idea of the variability within the taxonomic unit. When an existing taxon showed an excessive amount of variability among the fatty acid profiles of its samples, the taxon was further split into gas chromatographic subgroups (GC groups) to form separate library entries [34]. It should be noted that these GC groups have no real standing in bacterial taxonomy. MIDI suggests to assemble at least 10 different strains for the construction of a single entry in a proprietary identification library, but of course for some rare species this number might not be available. In general, we can state that each library entry gives rise to an $(n \times m)$ data matrix

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix}, \tag{5.3}$$

where m represents the number of features in the fatty acid peak naming table and n reflects the number of fatty acid profiles that were generated for the construction of the library entry. Each row of the matrix thus corresponds with a bacterial strain used for the library construction and each column is associated with a feature from the peak naming table. The value a_{ik} represents the relative area of the fatty acid peak (or peaks in case k is a summed feature) that contributes to feature k in the ith sample, with respect to the total area of all named peaks in the profile of sample i, as was defined in (5.2). To get an idea, the commercially available peak naming table of the TSBA50 method has 135 peak naming windows, assembled into $m = 123$ different features by the definition of 7 groups of summed features (see section B.3).

The identification problem raises the important question on how to make an intelligent estimation of the closeness of an unknown sample given by the row vector

$$x = (x_1, x_2, \dots, x_m), \tag{5.4}$$

representing its whole-cell fatty acid methyl ester profile encoded in a similar way as the rows composing the data matrix of a library entry, on the one hand, and on the other hand the taxonomic unit associated to a library entry with a data matrix as given in (5.3). A naive approach to solve this problem is to calculate the mean fatty acid profile from the patterns in the library entry as the row vector given by

$$\mu = (\mu_1, \mu_2, \dots, \mu_m), \tag{5.5}$$

with

$$\mu_k = \frac{1}{n} \sum_{i=1}^{n} a_{ik} \qquad 1 \leq k \leq m, \tag{5.6}$$

and to express the closeness of the unknown sample to the library entry as the *Euclidean distance* between the fatty acid profile x of the unknown sample and the mean fatty acid profile μ of the library entry in the following way

$$d^2(x, \mu) = \sum_{k=1}^{m} (x_k - \mu_k)^2. \tag{5.7}$$

We have indicated this as a naive measure because application of the Euclidean distance is not completely suitable for the purpose of identification based on fatty acid patterns, since the distance measure is isotropic and the identification problem is not, as every feature may not have the same behaviour. As an illustration, imagine that the patterns in a library entry show relative areas that are densely concentrated around t_k for fatty acid k, while for fatty acid l the same patterns are within a much wider interval around the value t_l. Then a difference d between x_k and t_k is much more significant than the same difference d measured between x_l and t_l. Euclidean distance does not take into account this possible asymmetry. The amount of variability for each fatty acid can be empirically determined in terms of the variance per feature over all samples in a library entry, represented as the row vector

$$\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_m)\,, \tag{5.8}$$

with

$$\sigma_k = \frac{1}{n}\sum_{i=1}^{n}(a_{ik} - \mu_k)^2 = \frac{1}{n}\sum_{i=1}^{n} a_{ik}^2 - \mu_k^2 \qquad 1 \leq k \leq m\,. \tag{5.9}$$

The closeness between the fatty acid profile x of the unknown sample and the mean fatty acid profile μ of the library entry, relative to the variance σ within the profiles of the library entry, can then be expressed as the *normalized Euclidean distance* defined by

$$d^2(x, \mu, \sigma) = \sum_{k=1}^{m} \frac{(x_k - \mu_k)^2}{\sigma_k}\,. \tag{5.10}$$

When the Euclidean distance or a related similarity coefficient is used, the major fatty acids (i.e. the fatty acids that occur in large quantities) account for most of the global similarity or dissimilarity, whereas the minor fatty acids, which may be as useful for differentiation, have little impact on the overall similarity [49]. This is exactly the problem that is resolved by the application of the normalized Euclidean distance for the comparison of unknown fatty acid patterns with the mean fatty acid profile of a library entry. However, this distance measure still treats the different fatty acids features as completely independent units, while in reality there might occur some transformations of the fatty acid composition due to small temperature shifts or age differences [30]. For example, 16:0 might turn into 16:1 due to the action of a desaturase [34]. These interrelations between the fatty acid compositions of a library entry can be captured within the $(m \times m)$ covariance matrix

$$\Sigma = \begin{bmatrix} c_{11} & c_{12} & \ldots & c_{1m} \\ c_{21} & c_{22} & \ldots & c_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \ldots & c_{mm} \end{bmatrix}\,, \tag{5.11}$$

where the covariance between two fatty acids k and l is empirically determined as

$$c_{kl} = \frac{1}{n}\sum_{i=1}^{n}(a_{ik} - \mu_k)(a_{il} - \mu_l)\,. \tag{5.12}$$

As such, the covariance matrix captures the mole-for-mole relationship of the conversion of one fatty acid into another. Note that for the diagonal elements of the covariance matrix

holds that $c_{kk} = \sigma_k$. Taking into account both the variances and covariances of the fatty acids found in the patterns of the library entry, the closeness between the fatty acid profile x of the unknown sample and the mean fatty acid profile μ of the library entry can be expressed using the standard formula for multivariate Gaussian (or normal) distance given by

$$d^2(x, \mu, \Sigma) = (x - \mu)\, \Sigma^{-1}\, (x - \mu)^{\mathrm{T}}\,. \tag{5.13}$$

Herein, the superscript in the expression $(x-\mu)^{\mathrm{T}}$ refers to the use of the transposed column vector of $(x - \mu)$, and Σ^{-1} is the inverse of the covariance matrix Σ. The matrix multiplication in (5.13) results in a single value, that is known as the *Mahalanobis distance*. In case the features are effectively independent of each other, the non-diagonal elements (covariances) of the covariance matrix are zero, and the Mahalanobis distance reduces to the normalized Euclidean distance. Additionally, when the fatty acids are isotropic, the covariance matrix becomes the identity matrix (except for a multiplicative constant) and the Mahalanobis distance is reduced to the usual Euclidean distance. But in general, the Mahalanobis distance is more sensitive than the Euclidean distance as it takes into account the variability and correlation among the fatty acids. Fatty acid profiles with equal Mahalanobis distance to the mean feature vector of a library entry form a multidimensional ellipsoid. The center of the ellipsoid is given by the mean fatty acid profile μ, and the direction of the axes is indicated by the eigenvectors of Σ. The length of the axes is given by $2\sqrt{\lambda_i}$, where λ_i ($1 \leq i \leq m$) represent the ith eigenvalue of Σ. In principle, at least 2^m different fatty acid profiles are required for the construction of a library entry wherefore the covariance matrix Σ is not singular (i.e. the determinant of the matrix is not zero). For most practical library entries it will thus be impossible to calculate the inverse of the covariance matrix directly. The procedure suggested in this case by a pattern recognition technique called SIMCA (Soft Independent Modeling of Class Analogy) is based on the eigenvalue-eigenvector analysis of the matrix Σ. A small value is added to all eigenvalues to avoid that some eigenvalues are zero or close to zero. To get the inverted matrix Σ^{-1}, the adjusted eigenvalues are then inverted and rotated by the eigenvector matrix. A similar approach is implemented in the Sherlock MIS (Charley Carter, Microbial ID Inc., personal communication).

The correlation between an unknown organism's fatty acid profile and a given library entry is then expressed as a value within the unit interval [0,1], termed as the *similarity index* (SI) within the context of the Sherlock MIS suite. Based on the Mahalanobis distance d calculated in (5.13), the SI of a particular fatty acid profile compared to the established population mean of the library entry is given by

$$\mathrm{SI}(x, A) = e^{-(\alpha d)^2}\,, \tag{5.14}$$

where α is equal to a constant such that SI = 0.6 when $d = 3.0$ [24]. These latter two values can be altered as the parameters of proprietary identification libraries, but usually these preset default values perform very well. A value of 1.0 for the similarity index means a perfect match with the taxon associated to the library entry. The identification procedure is completed by comparing the fatty acid profile of the unknown organism with all entries of the chosen identification library, and presenting a summary of the best matches. An example of this can be found at the bottom of the sample composition report shown in Figure 5.4, for the commercial identification library TSBA50 (note that peak naming

tables and identification libraries are essentially independent components of the Sherlock MIS, but both modules carry the same name TSBA50 in this particular case). The MIDI documentation suggests that strains with a similarity of 0.5 or higher and with a separation of 0.1 between the first and second match can be considered as good library identifications. If the similarity index falls within the interval [0.3,0.5] and is well separated from the second choice (> 0.1), the identification of the fatty acid profile may be acceptable but the unknown sample could be an atypical strain of the taxonomic unit associated to the library entry. Values lower than 0.3 suggest that the taxon of the unknown sample is probably not included in the identification library, but the best matched library entries anyhow indicate its closely related taxa [34].

5.2.5 Proprietary database construction

After chromatographic analysis, calibration, peak naming and library identification by the Sherlock MIS, each batch of fatty acid profiles is transfered to a relational Oracle 8.1.7 database management system (Oracle corporation, CA, USA). During this process, both the TSBA50 peak naming method and the TSBA50 identification library are chosen as the default options, but other combinations are optional when more specific types of data analysis are required. If a fatty acid profile has passed a redundancy check (all generated profiles are represented only once in the relational database), it is linked onto the integrated strain database as discussed previously in section 2.5. Remember that most mutual links with the integrated strain database can be established automatically, but that some ambiguous cross-references might need to be resolved by human intervention at a later stage. As such, the fatty acid methyl ester data becomes an integral part of the laboratory information management system (LIMS) of the Laboratory of Microbiology at the Ghent University and the associated BCCM™/LMG Bacteria Collection, where it forms a coherent unit with the integrated strain database, other empirically determined bacterial features, literature references and some administrative information. Many cross-reference links between the different components of the LIMS are either set up automatically or can be manually established. Moreover, as they are incorporated together with previous analyses into the integrated FAME database, the new fatty acid profiles become immediately available for further computational analysis within the BioNumerics software package (Applied Maths, Sint-Martens-Latem, Belgium) through application of the Database Sharing Tools module that implements the standard Open DataBase Connectivity (ODBC) protocol in the optional Connected Databases package. The choice of data management and data mining tools offered by the BioNumerics software package by far outperforms those that are built into the Sherlock MIS, and the data is readily accessible in a multi-user environment without the extra need for cumbersome manual data import procedures on behalf of the microbiologists. As the last step of this completely automated batch processing procedure, a personalized email message is dispatched to all staff members containing a summary of the Sherlock MIS identification results for the bacterial samples they have deposited into the batch. The curator of the FAME database, equally warned by an email notification after batch processing, can in parallel perform the required quality control checks on the fatty acid batch and take appropriate measures whenever necessary.

Since the very first recordings of whole-cell fatty acid methyl ester chromatograms in Februari 1989 using the Sherlock MIS, over fifteen years later, 52284 fatty acid profiles have accumulated in the FAME database that is collectively established by the Laboratory of Microbiology at the Ghent University and the BCCM™/LMG Bacteria Collection, with a total amount of 965796 peaks detected within these chromatograms. The majority of taxa covered within this vast knowledge base reflects the general interest of the laboratory in the taxonomy of aerobic bacteria, primarily isolated from environmental (non-clinical) samples. Most strains were subjected to fatty acid analysis in the framework of taxonomic research projects, during the initial screening stages for the polyphasic classification [43] of large sets of samples from the genera *Aeromonas* [13, 14, 15], *Arcobacter* [42], *Bordetella* [40], *Flavobacterium* [2], *Pseudomonas* [41], *Rhizobium* [38], *Streptomyces* [27], and *Xanthomonas* [47, 48, 49, 50], among many others. Chromatographic analysis of the fatty acid composition of a large group of strains isolated from Arctic and Antarctic waters revealed members of new and old taxa related to the genera *Alteromonas*, *Cytophaga*, *Glaciecola*, *Halomonas*, *Pseudoalteromonas*, *Rhizobium*, *Rhodococcus*, *Shewanella* and *Sulfitobacter* [28, 45]. It was found that in order to maintain fluidity of the membranes under low temperature conditions, polar isolates are characterized by high amounts of unsaturated fatty acids [28]. Microbial identification based on fatty acid analysis was also implemented for investigating the biodiversity of heterotrophic bacteria colonizing mural paintings that showed visual deterioration by microorganisms, uncovering the dominant presence of *Arthrobacter*, *Bacillus*, *Paenibacillus*, *Micrococcus* and *Staphylococcus* species, but also nocardioform actinomycetes and gram-negative bacteria [10]. Apart from its use as a fast screening technique, knowledge of the cellular fatty acid composition also plays an important role in the description of new bacterial taxa in many scientific publications. As an illustration, we refer to the descriptions of *Arcobacter butzleri* and *A. skirrowii* [42], *Brachybacterium fresconis* and *B. sacelli* [11], *Flavobacterium hydatis* [2] and *Leeuwenhoekiella aequorea* [31]. As a third application, the BCCM™/LMG Bacteria Collection also produces fatty acid profiles within the implementation of a total data quality management (TDQM) system, given the cheapness, high throughput and long-term reproducibility of the technique. Of course, complete information about the different contributions to the accumulation of the diverse range of fatty acid profiles in the database, would result in a sheer endless enumeration of scientific studies and other applications.

To get an impression of the high variation in the taxonomic units covered within our proprietary FAME database, an excerpt of the most dominant bacterial genera represented in the fatty acid database is depicted in Table 5.1. For all taxa incorporated within this list, the frequency of occurrence was estimated by extracting the identification interpretations from two autonomous information sources. The number of profiles associated to a given genus, as indicated in the column with header `LMG`, was determined by restricting the FAME database to the samples that are linked to strains that are deposited into BCCM™/LMG Bacteria Collection, either directly or by use of a synonym strain label as derived from the connection with the integrated strain database. This estimation is based on a restricted fraction of the database, but the accurracy of the strain identification is highly reliable, as most of the strains are characterized based on a polyphasic approach. The alternative frequency count included in the column with header `MIS`, is based on an interpretation that covers all samples of the FAME database, estimated by extracting the best match from the

genus	LMG	MIS	genus	LMG	MIS	genus	LMG	MIS
Bacillus	942	5667	*Microbacterium*	63	625	*Thermus*	1	223
Aeromonas	624	3789	*Geobacillus*	32	594	*Sphingomonas*	72	196
Pseudomonas	1024	3584	*Chryseobacterium*	97	577	*Flavobacterium*	146	195
Xanthomonas	2626	2415	*Rhodococcus*	46	546	*Paucimonas*	10	195
Vibrio	596	2014	*Micrococcus*	45	544	*Pseudoxanthomonas*	1	194
Stenotrophomonas	578	1584	*Arthrobacter*	99	532	*Sphingobacterium*	73	192
Staphylococcus	90	1548	*Brevundimonas*	99	442	*Sphingopyxis*	33	180
Paenibacillus	346	1224	*Enterococcus*	276	432	*Brevibacterium*	22	173
Burkholderia	412	1128	*Variovorax*	4	379	*Achromobacter*	63	170
Salmonella	15	977	*Pantoea*	159	376	*Ochrobactrum*	18	167
Corynebacterium	84	917	*Photobacterium*	66	346	*Klebsiella*	7	160
Acidovorax	86	858	*Novosphingobium*	5	327	*Listeria*	71	156
Acinetobacter	319	847	*Kurthia*	4	312	*Listonella*	55	156
Pseudoalteromonas	93	839	*Psychrobacter*	29	311	*Xanthobacter*	1	150
Brevibacillus	131	768	*Shewanella*	72	309	*Mycobacterium*	37	148
Microbacterium	63	625	*Kocuria*	35	282	*Yersinia*	17	140
Geobacillus	32	594	*Ralstonia*	70	259	*Lysobacter*	1	137
Chryseobacterium	97	577	*Lactobacillus*	52	256	*Paracoccus*	12	135
Rhodococcus	46	546	*Neisseria*	21	242	*Pectobacterium*	152	132
Micrococcus	45	544	*Enterobacter*	24	226	*Gluconobacter*	26	130
Arthrobacter	99	532	*Escherichia*	15	224	*Rhodobacter*	1	129
Brevundimonas	99	442	*Zobellia*	3	224	*Kluyvera*	2	124

Table 5.1: Overview of the most dominant genera within the FAME database. The number of profiles associated to a given genus, as indicated in the column with header `LMG`, was determined by restricting the FAME database to the samples that are linked to strains that are deposited into BCCM™/LMG Bacteria Collection. The alternative frequency count included in the column with header `MIS`, is estimated by extracting the best match from the identification against the Sherlock MIS TSBA50 identification library.

identification against the Sherlock MIS TSBA50 identification library. This identification is generally less accurate, as it solely relies on the interpretation of the fatty acid profiles, but it covers a broader range of the samples in the FAME database. In total, 1097 validly described taxa are at least represented by the fatty acid composition of their type strain in the FAME database, whereas several other reference strains have been scanned as well for most taxa. This amount significantly outnumbers the 888 library entries that are incorporated into the commercially available TSBA50 identification library of the Sherlock MIS, some of them even representing (artificial) GC groups of the same taxon. However, it is important to note that both knowledge bases only share 664 taxa, essentially making them valuable complementary information sources.

5.2.6 Data warehousing for OLAP

In the next two sections, we will present some more statistics of the proprietary FAME database of the Laboratory of Microbiology at the Ghent University and the BCCM™/LMG Bacteria Collection, and indicate how the information that is stored into this database can be exploited to improve the peak naming of the fatty acid chromatograms and to enhance the power of fatty acid analysis for the identification of unknown bacteria. However, before performing any further computational analysis on the large assembly of FAME database entries, the transactional information captured along the normalization principles of relational databases [5] was predigested in terms of the data warehousing paradigm, as to improve the

overall performance of the calculations involved in online analytical processing (OLAP). This data transformation included the introduction of redundancy into the physical storage of the data and the massive use of column indexing [17, 20].

In order to allow instant monitoring of the data quality for newly generated fatty acid profiles and estimate the reproducibility of whole-cell fatty acid analysis over a long period of time, most batch runs contained a reference sample of the *Stenotrophomonas maltophilia* type strain LMG 958^T ($\equiv$ ATCC 13637^T). Fatty acid profiles that failed to pass our propriety quality control checks in addition to the quality control performed by the Sherlock MIS, were discarded from the analytical data warehouse during this preprocessing stage. As a result, the total number of profiles taken into account for statistics was reduced to 49017, amounting for 940602 fatty acid peaks detected on the chromatograms.

5.3 Qualitative FAME analysis

5.3.1 Distribution of bacterial fatty acids

Qualitative analysis of the bacterial whole-cell fatty acid methyl ester composition solely relies on the presence or absence of certain fatty acids, without taking into account absolute or relative amounts of the compounds detected within the bacterial cell. As a first qualitative application, we were interested in the distribution of the different types of fatty acid molecules encountered in the broad diversity of environmental aerobic bacteria as a whole, which is representatively covered by the samples in our proprietary FAME database. To this end, Figure 5.5 shows the histogram of chromatographic fatty acid peaks that were named using the TSBA50 peak naming method, derived from the reduced set of high-quality profiles in the FAME database. The named fatty acids are depicted in descending order of occurrence within the chromatograms.

This histogram clearly demonstrates that the different chemical constellations of fatty acids that can be detected by the Sherlock MIS are not equally distributed among the aerobes. If occurrence in 25% of the chromatographic profiles within the database is taken as the cut off level, we can conclude that the straight chain fatty acids with 12 to 18 carbon atoms (except for 13:0) and the iso-branched fatty acids with 13 to 17 carbon units are omnipresent in the aerobic bacteria, together with 15:0 anteiso and 17:0 anteiso, the 3-hydroxy fatty acids 12:0 3OH and 14:0 3OH (as the dominant component of summed feature 2), and the unsaturated fatty acids 16:1 ω7c (as the dominant component of summed feature 3), 17:1 $\omega 8c$ and 18:1 ω7c. The straight chain fatty acid 16:0 was even found in 95% of the generated profiles, and is thus qualitatively the least discriminatory fatty acid. On the opposite side of the histogram we find the very rare but highly discriminatory fatty acids, *in casu* the evenly numbered anteiso fatty acids 12:0 anteiso (specifically encountered in *Flammeovirga aprica, Flavobacterium saccharophilum* and *Marinilabilia salmonicolor*), 14:0 anteiso (specifically encountered in *Alicyclobacillus acidoterrestris* and *Tenacibaculum maritimum*) and 16:0 anteiso (specifically encountered in *Streptomyces rutgersensis*

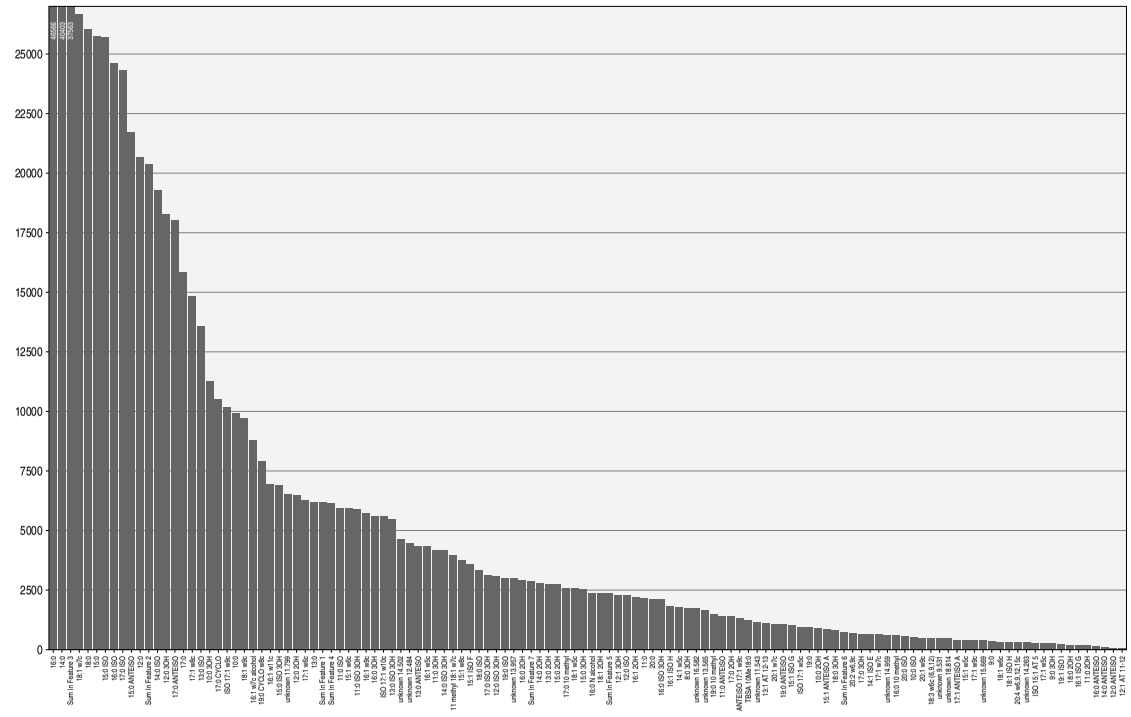


Figure 5.5: Histogram showing frequency of occurrence for the fatty acid peaks named with the TSBA50 peak naming method that are covered within the proprietary FAME database.

subsp. *rutgersensis*, *Streptomyces septatus* and *"Streptomyces tenebrarius"*). Remark that the use of the double quotes in the latter case means that the name of the taxon has not been validly published. It is quite strange that although these rare fatty acids were included as naming windows in the TSBA50 peak naming method, none of the entries in the associated TSBA50 identification library showed any traces of these fatty acids in their average peak profiles. The same observation could be made for the fatty acids compounds 14:1 iso E (specifically encountered in *Moritella marina* and *Vibrio logei*), 17:1 ω5c (specifically encountered in *Streptomyces limosus*, *Streptomyces murinus*, *Streptomyces odorifer* and *Streptomyces rimosus* subsp. *rimosus*) and 18:0 2OH (specifically encountered in *Aeromicrobium erythreum* and *Aeromicrobium fastidiosum*), which also seldom occur in the strains covered in our FAME database. Apparently, all chromatographic fatty acid peaks that appeared within these naming windows did not pass the quality threshold used for the construction of the Sherlock MIS TSBA50 identification library, and were discarded as zero features. We will come back to this quality threshold applied by MIDI in subsection 5.4.1. None of the taxa that contained traces of these rarely occurring fatty acid compounds in our proprietary FAME database, except for the species *Moritella marina* and *Vibrio logei*, were incorporated for the construction of the TSBA50 identification library.

5.3.2 Uniqueness of fatty acid combinations

In the description of bacterial taxa within the specialized scientific literature, one can often find statements about the combination of several fatty acids that is supposed to be qualitatively unique for a specific taxonomical unit. Such hypothesis can be easily tested against the complete database of fatty acid profiles, using straightforward SQL queries [5]. As an example, we reconsider the claim by Yang *et al.* [50], based on their observation that the methyl-branched fatty acid 11:0 iso, together with the two branched-chain hydroxy fatty acids 11:0 iso 3OH and 13:0 iso 3OH occurred in almost all *Xanthomonas* strains within their extensive set of samples. Accordingly, the authors state that to their knowledge, these three fatty acids have not been found together in other bacteria and are thus useful features to differentiate *Xanthomonas* from other bacteria. When subjected to the FAME database, it becomes immediately clear that apart from being unique to *Xanthomonas* and *Stenotrophomonas* (a genus split off from *Xanthomonas* after publication of Yang *et al.* [50]), this combination of fatty acid compounds is also found in some or all strains of the species *Burkholderia cepacia*, *Fulvimonas soli*, *Idiomarina baltica*, *Pseudomonas beteli*, *P. boreopolis*, *P. hibiscicola*, *P. pictorum*, *Rhodovulum sulfidophilum*, *Shewanella denitrificans* and *S. frigidimarina*. It should however be noted that on the basis of a large set of phenotypic features, *Pseudomonas beteli* and *P. hibiscicola* were found to be synonyms of *Stenotrophomonas maltophilia* [44]. This has not yet been confirmed by solid genotypic evidence, but at least one can suppose that these species are likely to be closely related. As such, the original claim should be relaxed in some sense. When taking into account quantitative amounts of the fatty acid content of the bacterial cells, these different taxa can be easily separated from the *Xanthomonas-Stenotrophomonas* group on the basis of the three previously named fatty acids. As a general conclusion, quantitative combinations of fatty acid compounds are rarely exclusive for certain taxa, even for higher order branches in the hierarchy, in a sense that exceptions can be found in most cases. This once more underscores the necessity for the application of qualitative measures in order to improve the separation of bacterial taxa based on their fatty acid content.

5.3.3 Delineation of peak naming windows

In order to clearly situate this final type of qualitative analysis, let us first inspect the Sherlock MIS fatty acid composition report of strain LMG 21428^T shown in Figure 5.6. This is the type strain of *Pseudoalteromonas prydzensis*, a psychrotrophic (i.e. thriving at relatively low temperatures), halotolerant (i.e. thriving at relatively high salt concentrations) bacterium isolated from the Antarctic sea ice [3]. Remark that no correct identification at the species level was accomplished by the commercial TSBA50 identification library, because no entry representing this taxon is currently incorporated within the identification library. However, the exact genus identification was found anyhow, indicated by library matches with some closely neighbouring species of the genus *Pseudoalteromonas*. Also note the illustration of the summed feature approach, where the relative amounts for the peaks 16:1 ω7c and 15:0 iso 2OH, bundled into summed feature 3, are added. As a last minor comment on this fatty acid composition report, we point out the fact that dur-

```
Sherlock Version: 3.10                                                    DATA7:E02429367A    31-MAR-04 18:20:42

-------------------------------------------------------------------------------------------------------------------
ID:     6583       PSEA-PRYDZ(LMG21428T/QC3/02/Q3/20C/P18)                      Date of run: 29-APR-02 11:14:34
Bottle: 6          SAMPLE      [TSBA40]                                         Date edited: 30-APR-02 13:31:47

   RT      Area    Ar/Ht Respon     ECL          Name            %        Comment 1             Comment 2
------- --------- ------ ------   ------  -------------------- ------  --------------------  --------------------
  1.666 443829656  0.028  . . .    6.994  SOLVENT PEAK . . . .  . . .  < min rt
  2.517       169  0.020  . . .    8.826   . . . . . . . . . .  . . .  < min rt
  3.701       284  0.031  1.066   10.999  11:0 . . . . . . . .   0.13  ECL deviates -0.001   Reference -0.001
  4.063       630  0.030  1.049   11.425  10:0 3OH . . . . . .   0.28  ECL deviates  0.003
  4.382      2677  0.029  1.035   11.801  unknown 11.799 . . .   1.18  ECL deviates  0.002
  4.552      4049  0.031  1.028   12.001  12:0 . . . . . . . .   1.78  ECL deviates  0.001   Reference  0.002
  4.648       462  0.028  1.025   12.090  11:0 ISO 3OH . . . .   0.20  ECL deviates  0.001
  5.021      3981  0.033  1.015   12.438  11:0 3OH . . . . . .   1.72  ECL deviates  0.000
  5.390      2695  0.032  . . .   12.782   . . . . . . . . . .  . . .
  5.624      1140  0.038  1.000   13.000  13:0 . . . . . . . .   0.49  ECL deviates  0.000   Reference  0.002
  5.750      2527  0.036  0.998   13.098  12:0 ISO 3OH . . . .   1.08  ECL deviates  0.000
  5.992       393  0.038  0.993   13.286  12:1 3OH . . . . . .   0.17  ECL deviates -0.002
  6.208     18230  0.036  0.990   13.454  12:0 3OH . . . . . .   7.70  ECL deviates -0.000
  6.671      3049  0.037  . . .   13.814   . . . . . . . . . .  . . .
  6.910      4016  0.037  0.980   14.000  14:0 . . . . . . . .   1.68  ECL deviates -0.000   Reference  0.002
  7.332       605  0.038  . . .   14.287   . . . . . . . . . .  . . .
  7.600      2326  0.039  0.972   14.471  Sum In Feature 1 . .   0.97  ECL deviates  0.001   13:0 3OH/15:1 i I/H
  8.072     11259  0.040  0.968   14.793  15:1 w8c . . . . . .   4.65  ECL deviates -0.000
  8.165       930  0.039  0.967   14.856  15:1 w6c . . . . . .   0.38  ECL deviates  0.000
  8.376     12287  0.038  0.966   15.000  15:0 . . . . . . . .  . . .  ECL deviates  0.000
  8.814      3847  0.043  . . .   15.274   . . . . . . . . . .  . . .
  9.156       652  0.047  0.961   15.488  Sum In Feature 2 . .   0.27  ECL deviates -0.000   14:0 3OH/16:1 ISO I
  9.379      1173  0.040  0.960   15.627  16:0 ISO . . . . . .   0.48  ECL deviates -0.000   Reference  0.002
  9.614      2818  0.033  0.958   15.774  16:1 w9c . . . . . .   1.15  ECL deviates  0.000
  9.684     79430  0.043  0.958   15.818  Sum In Feature 3 . .  32.47  ECL deviates -0.004   16:1 w7c/15 iso 2OH
  9.737     10748  0.030  0.958   15.851  Sum In Feature 3 . .   4.39  ECL deviates -0.001   15:0 ISO 2OH/16:1w7c
  9.974     39721  0.042  0.957   15.999  16:0 . . . . . . . .  16.22  ECL deviates -0.001   Reference  0.000
 11.032      1032  0.040  0.953   16.630  17:0 ISO . . . . . .   0.42  ECL deviates -0.000   Reference  0.001
 11.190       545  0.039  0.953   16.724  17:0 ANTEISO . . . .   0.22  ECL deviates  0.001   Reference  0.002
 11.305     27800  0.044  0.953   16.793  17:1 w8c . . . . . .  11.30  ECL deviates  0.001
 11.379      4596  0.055  . . .   16.837   . . . . . . . . . .  . . .
 11.654      7540  0.045  0.952   17.001  17:0 . . . . . . . .   3.06  ECL deviates  0.001   Reference  0.002
 13.067     17560  0.048  0.949   17.823  18:1 w7c . . . . . .   7.11  ECL deviates -0.000
 13.372       811  0.048  0.949   18.000  18:0 . . . . . . . .   0.33  ECL deviates -0.000   Reference  0.000
 13.513       416  0.035  0.949   18.082  11 methyl 18:1 w7c .   0.17  ECL deviates  0.001
 14.743       886  0.044  . . .   18.798   . . . . . . . . . .  . . .
*******      2326  . . .  . . .    . . .  SUMMED FEATURE 1 . .   0.97  15:1 ISO H/13:0 3OH   13:0 3OH/15:1 i I/H
******* . . . . .  . . .  . . .    . . .   . . . . . . . . . .  . . .  15:1 ISO I/13:0 3OH
*******       652  . . .  . . .    . . .  SUMMED FEATURE 2 . .   0.27  12:0 ALDE ?           unknown 10.928
******* . . . . .  . . .  . . .    . . .   . . . . . . . . . .  . . .  16:1 ISO I/14:0 3OH   14:0 3OH/16:1 ISO I
*******     90178  . . .  . . .    . . .  SUMMED FEATURE 3 . .  36.87  16:1 w7c/15 iso 2OH   15:0 ISO 2OH/16:1w7c

Solvent Ar  Total Area  Named Area  % Named  Total Amnt  Nbr Ref  ECL Deviation  Ref ECL Shift
----------  ----------  ----------  -------  ----------  -------  -------------  -------------
 443829656      271116      243152    89.69      234348       10          0.001          0.001
-------------------------------------------------------------------------------------------------------------------
          TSBA50 [Rev 5.0] Pseudoalteromonas  . . . . . . . . . . . . . . . . 0.655
                           P. nigrifaciens  . . . . . . . . . . . . . . . . 0.655
                           P. tetraodonis . . . . . . . . . . . . . . . . . 0.651
                           P. haloplanktis  . . . . . . . . . . . . . . . . 0.560 (Vibrio, Alteromo
                             P. h. haloplanktis . . . . . . . . . . . . . . 0.560 (Vibrio, Alteromo
```

Figure 5.6: Sherlock MIS fatty acid composition report for the *Pseudoalteromonas prydzensis* type strain LMG 21428^{T}.

ing transition from the commercial peak naming method TSBA40 to peak naming method TSBA50, MIDI considered to treat the straight chain fatty acid compound 15:0 as a zero feature, in order to avoid artificial variance in the fatty acid profiles that results in poor similarity index calculations. *Id est*, although the chromatographic peak 15:0 is still identified by the peak naming method TSBA50, it is no longer taken into account during calculation of the relative amounts of the fatty acid compounds. This decision is supported by MIDI following work with coryneforms and related organisms, which often produce unknown peaks located in the 15:0 naming window that are non-reproducible fragments of long chain mycolic acids, and the fact that acid-fast organisms often produce fragments that also fall within the 15:0 naming window, although they are not related to this fatty acid compound (Ralph Paisley, Microbial ID Inc., personal communication). MIDI also claims that the number of bacteria containing fatty acid compound 15:0 is rather low, but this was clearly contradicted by our investigation on the distribution of bacterial fatty acids

in subsection 5.3.1, where a 15:0 peak was found in 52.5% of all fatty acid profiles. Consequently, the 15:0 compound (or the breakdown product that falls within this window) is no longer taken into account for the calculation of the total amount of named peaks, resulting in a fraction of $100 \times \frac{243152}{271116} = 89.7\%$ of the total fatty acid content being recovered by the Sherlock MIS for further computational analysis.

Most striking, however, is the observation that the Sherlock MIS has failed to name some of the peaks in the chromatogram of *Pseudoalteromonas prydzensis* LMG 21428^T, although some of these unnamed fatty acids are present in rather traceable amounts. The obvious reason for this hiatus, is that the peak naming windows of the TSBA50 method do not cover the complete ECL range within the interval [9.000,20.000]. Outside of this interval, no peaks are named by the Sherlock MIS for the aerobic bacteria, due to the limited length of the capillary column. During their analysis of the cellular fatty acids of *Agrobacterium*, *Bradyrhizobium*, *Mesorhizobium*, *Rhizobium* and *Sinorhizobium* species using the Sherlock MIS, Tighe *et al.* stated the important note that when evaluating the fatty acid composition of large groups of previously unexamined strains, the detection of new compounds is not uncommon and it is important to include them into the peak naming methods for establishing accurate relationships between groups [38]. We will demonstrate that this is a truism that has been overlooked all too often. Vandamme *et al.* [42] also listed some unnamed peaks during a polyphasic study of the genus *Arcobacter*, most of them now being included into the commercial TSBA50 peak naming method.

Inspired by these observations, we have further investigated the success rate of peak naming in the complete database of fatty acid profiles. In total, 781456 peaks (83%) have been named by the TSBA50 peak naming method, whereas 159146 peaks remained unnamed, from which 99946 peaks (11%) fall within the ECL range [9.000,20.000]. During forthcoming calculations, we have primarily concentrated ourselves on the peaks that fall within this latter ECL interval. In order to enable differentiation of the chromatographic peaks representing reproducible fatty acid compounds from the artefacts of the technique, we have scrutinized the distribution of chromatographic peak locations by calculating a histogram showing the occurrence of chromatogram peaks grouped by their normalized ECL position. Sherlock MIS fatty acid composition reports show ECL values with a precision of 10^{-3}, so we have stuck to histogram bins of 0.001 ECL units wide. A graphical representation of the histogram, covering the complete ECL range [7.000,21.000] of fatty acid methyl ester peak locations detected within the complete database of chromatograms, is depicted in Figure 5.7. The contributions of fatty acid peaks named by the TSBA50 peak naming method of the Sherlock MIS are indicated in green, while the unnamed fatty acid peak contributions are indicated in red. Remark that due to rounding errors in the representation of the ECL values, some histogram bins may contain both named and unnamed peak contributions. In these cases, we have plotted the contribution of the unnamed peaks on top of those of the named peaks. At a glance, one easily notices by analogy with the histogram peaks that represent the named fatty acid features, that quite a large number of unnamed peaks are scattered in a non-random way throughout the histogram, representing clusters of chromatographically inseparable fatty acid compounds ranging in abundance from high, over moderate, to low. These clusters are good candidates to become part of the peak naming process, by inclusion of their corresponding ranges as naming windows into

the peak naming methods.

More can be learned about the distribution of the large set of fatty acid chromatographic peaks by heavily zooming in the scope towards a limited section of the ECL range. An example of the peak occurrence histogram restricted to the ECL interval between 18.600 and 19.000 is given in Figure 5.8. Appendix B contains the complete list of histogram illustrations, covering the entire ECL range [8.000,21.000] in chunks of 0.200 ECL units. The naming windows of the TSBA50 peak naming method are incorporated at the bottom of the histogram, by means of a line with arrows at the outsides (only one arrow is shown if the corresponding window bridges across the boundaries of the selected interval), with the name of the associated fatty acid compound underneath. Overlapping windows can be easily discriminated and are shown at different vertical positions to improve readibility. This representation has turned out to be an utmost handy vehicle for evaluating the delineation of peak naming windows, from which a number of important conclusions can be drawn.

First of all it should be clearly stated that in general there is a good correspondence between the peaks in the histogram on the one hand, and the naming windows as defined by the TSBA50 peak naming method of the Sherlock MIS on the other hand. This underscores the assumption that robust histogram peaks represent identical fatty acid compounds that are more or less abundantly present in cells of aerobic bacteria. However, in addition the histogram also reveals some clearcut problems with the delineation of peak naming windows which urgently need to be sorted out. Three different situations occur.

Some histogram peaks only partially overlap with the naming windows as defined by the Sherlock MIS. These histogram peaks suggest that the ECL delineation of several naming windows should be redefined, requiring a shift for some of the naming windows, an increase or decrease in the width of these naming windows, or a complete restructuration of some neighbouring naming windows. Figure 5.8 contains an illustration of this issue in the definition of summed feature 7, combining the three naming windows 19:1 ω6c, 19:0 cyclo ω10c and unknown 18.846. From the histogram we can infer that there are probably only two fatty acids within the range of summed feature 7, and that some extension of the window range towards the left is required in order to cover the leftmost histogram peak. Moreover, these fatty acid peaks are apparently separated quite nicely by the chromatograph, so that there is perhaps even no need for the introduction of a summed feature in this particular case. A similar situation occurs for the combination of 12:0 aldehyde and unknown 10.928 into summed feature 2, which may probably be resolved by merging these two windows into a single naming window that represents 12:0 aldehyde. Many more examples can be spotted along the tracks of the detailed histogram shown in Appendix B.

Secondly, for several naming windows, one or more shoulders appear within the corresponding histogram range. This situation most probably reflects the interference of multiple fatty acids that cannot be clearly separated due to the limitations of the chromatographic technology. Representing these cases by a single naming window is not harmfull for the bacterial identification system itself, which also makes use of summed features to cope with overlapping naming windows. However, in light of a consistent application of the summed feature approach, it would be rather natural to sort out each of these cases by the

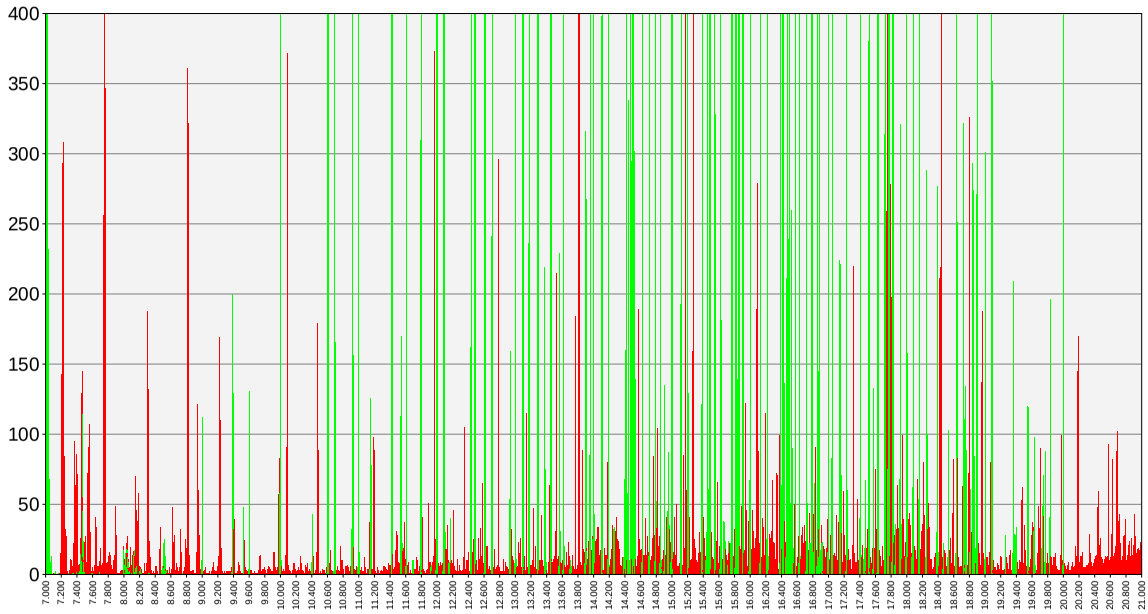


Figure 5.7: Complete histogram of fatty acid methyl ester peak locations detected within the chromatograms. The contributions of the fatty acid peaks named by the TSBA50 peak naming method of the Sherlock MIS are indicated in green, while the unnamed fatty acid peak contributions are indicated in red.

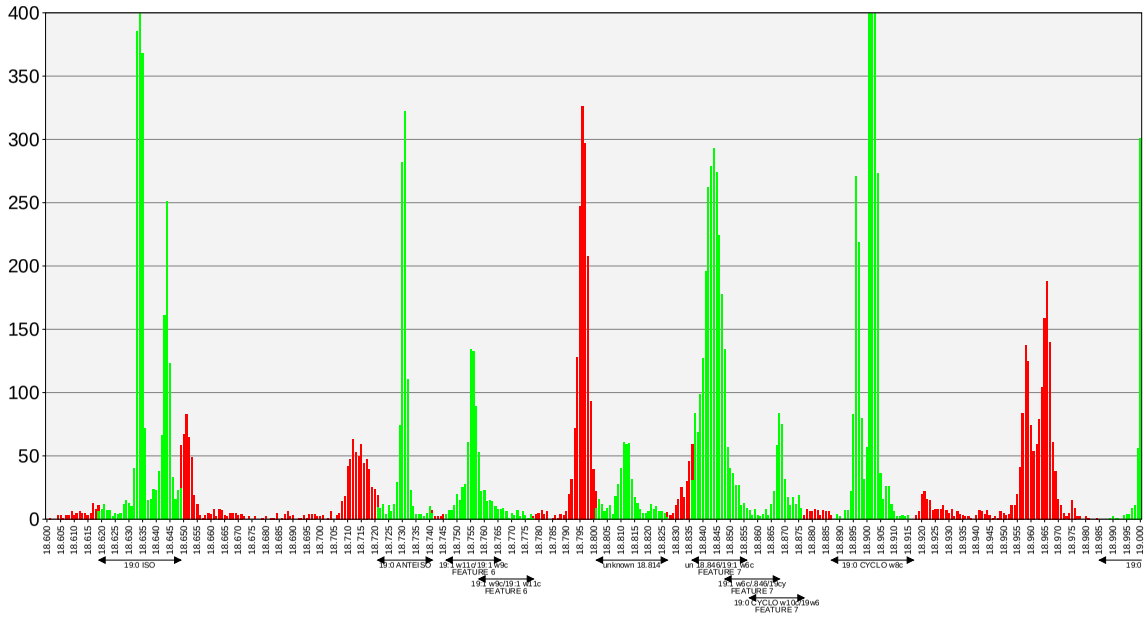


Figure 5.8: Histogram of the fatty acid methyl ester peak occurrences between ECL 18.600 and ECL 19.000. The contributions of the fatty acid peaks named by the TSBA50 peak naming method of the Sherlock MIS are indicated in green, while the unnamed fatty acid peak contributions are indicated in red. The peak naming windows are indicated below the histogram.

introduction of several overlapping naming windows constituting a single summed feature. This would clearly improve the descriptive power of the fatty acid identification system, as the different chromatographic peaks could be more precisely named with the most likely fatty acid compound that they represent. In the detailed region of the histogram covered by Figure 5.8, the window named 19:0 iso evidently shows three spikes, suggesting that the naming window should better be replaced by a summed feature with three overlapping naming windows. *Idem dito* for the window representing the fatty acid 19:0 cyclo ω6c, which could be separated into two or three overlapping windows along the information provided by the histogram. The peak named 14:1 ω5c used to be a summed feature in the AEROBE method (a predecessor of the TSBA50 method), composed of two naming windows representing 14:1 ω5t and 14:1 ω5c. A similar duality is suggested by the peak occurrence histogram calculated from our proprietary database, supporting the reinstatement of this summed feature. Again, much more examples of analogous cases can be easily spotted in the other parts of the histogram shown in Appendix B.

Finally, some red coloured histogram peaks immediately catch the eye, representing fatty acid compounds that currently have no overlap with any existing naming window of the TSBA50 peak naming method. Consequently, those peaks are not taken into account during the bacterial identification procedure implemented by the Sherlock MIS, as was previously illustrated by the fatty acid composition report in Figure 5.6. An investigation on the ECL delineation of these histogram peaks and their inclusion as naming windows in the existing peak naming methods could seriously enhance the discriminatory power of the bacterial identification system that makes use of named peaks, while the resolution by mass spectroscopy of the fatty acid compound that corresponds with those histogram peaks would simultaneously improve the naming power of the fatty acid identification system for the description of bacterial taxa. In the framework of this chapter we have primarily focused our attention onto the delineation of the most abundant and clearly separated unnamed peaks showing up in the histogram, and scrutinized their significance for the taxa that are included in the FAME database that was discussed in section 5.2. As an immediate result, 32 new naming windows have been established, accounting for an increase of 26% in the number of features recognized by the fatty acid analysis. The fact that these additional fatty acids are only present in a moderate or low amount of the samples screened within the FAME database, could indicate that they constitute important discriminatory features for certain bacterial taxa, both in a quantitative and qualitative manner. We come back in more detail on some of the quantitative aspects of the newly detected peaks in subsection 5.4.1. The correct naming of the compounds associated with the newly identified histogram peaks is postponed until some future point in time.

Some evident parallels can be drawn between the manual delineation of the fatty acid naming windows using a histogram of normalized chromatographic peak locations, as it was discussed in this subsection, and the histogram-based multiple band matching method discussed in subsection 4.7.2. Both methods calculate a histogram based on the peak positions of large sets of banding patterns for the detection of homologous peaks and use overlapping positional windows to accomodate for small peak shifts due to technical imperfections. However, the fatty acid peak recognition approach differs in its application of variable sized windows instead of a fixed position tolerance ε and the introduction of

summed features for the agglomeration of peaks in overlapping windows. These characteristics might also prove their value for the discretized comparison of particular types of gel electrophoresis patterns.

5.4 Quantitative FAME analysis

Besides the presence of unique fatty acids, quantitative information can be used as well to enhance the differentiation between taxonomic units. Many authors have reported the equality of the qualitative content of some closely related microbial taxa, whereas these same groups of bacteria could be easily discriminated when taking into account the relative distribition of the fatty acid compounds. Huys *et al.* [13] concluded that the differences between species of the genus *Aeromonas* were mainly quantitative, which suggests that differentiation of the genotypic hybridization groups in the genus *Aeromonas* by fatty acid methyl ester analysis is possible only when a high-resolution identification system like the Sherlock MIS is used. They report that although the fatty acid profiles were very similar, minor quantitative variations could be used to differentiate phenospecies and/or hybridization groups. Quasi identical conclusions were drawn from the fatty acid composition of the genus *Bordetella*, where *B. bronchiseptica* strains were distinguished from all other strains by small but significant amounts of fatty acids 12:0 2OH, 19:0 cyclo ω8c, and summed feature 7, proving again that, in general, strains belonging to the same species exhibit only minor quantitative differences in their fatty acid profiles [40].

Vancanneyt *et al.* [41] claimed that differences in the growth conditions and extraction procedures may influence the quantitative behaviour of the fatty acid analysis rather than the qualitative content of the fatty acid composition of the bacterial cell. This underscores the importance of the well-designed incubation protocols that were strictly followed as much as possible for the construction of our in-house fatty acid database. Variations for the most abundant fatty acids were generally found to be less than 3% among repeated profiles of the same strain, indicating the good reproducibility of the standardized procedure [50]. We will now demonstrate how the knowledge covered into our proprietary database can be exploited as a good alternative for the identification of unknown bacterial strains and how the same information can endorse the significance of some newly detected fatty acid peak naming windows for the bulk identification of the chemical structure corresponding with chromatogram peaks.

5.4.1 Stability of new fatty acid peaks

In order to validate the new naming windows that were delineated from the chromatographic peak occurrence histogram in subsection 5.3.3, it is required to check whether they correspond with *stable* fatty acids. Used in this context, stable means that for many well-characterized strains of a particular organism, a chromatographic peak for the extracted fatty acid must appear in a significant amount in most of the sample profiles [22]. This

stability could already be predicted as a general property of the naming windows from the evident non-random distribution of the selected histogram peaks, but in this section we will further scrutinize the importance of the newly delineated peaks by investigating their significance for all taxa at the different levels in the taxonomic hierarchy.

To attain reliable and well-founded groupings of taxonomic units required for this kind of data mining activity, it is of utmost importance to have a good idea about the identification of the biological samples included in the FAME database that was discussed in section 5.2.5. Because most fatty acid profiles were included in the database during large screenings of new bacterial isolates within the framework of taxonomic research projects, clearcut polyphasic strain identifications were not always available at the time of encoding the samples into the database. As to assure the necessary feedback, the fatty acid records were automatically or manually linked onto the integrated strain database, according to the flexible cross-referencing approach discussed in section 2.5. The identification information of the integrated strain database is independently updated as improved taxonomic insights are gained from new experimental evidence for the bacterial strains. In order to get full confidence in the characterization of the bacterial strains used for determining the relationship between existing taxa and the new fatty acid peaks, we further restricted the scope of the calculations to the fatty acid profiles of strains that were taken from the BCCM™/LMG Bacteria Collection at the time of chromatographic fatty acid analysis or were deposited into that collection at a later stage. After all, bacterial strains are only incorporated into the BCCM™/LMG Bacteria Collection after they are identified according to an extensive polyphasic analysis, and their identification is constantly monitored against and updated according to the most recent insights in bacterial taxonomy. It should be noted that, in general, at the time of fatty acid analysis most strains are not yet deposited into the culture collection but are included during the final stages of the identification process, which goes along with relabelling of the bacterial samples by the assignment of an LMG number. This once more underscores the importance of cross-referencing the FAME database with the integrated strain database by means of the approach reviewed in section 5.2.5. In total, 14548 (30%) of the fatty acid profiles that were linked to the integrated strain database could be traced as being deposited into the BCCM™/LMG Bacteria Collection. One lacuna still present in the implementation of the integrated strain database is the lack of an automatic tool for resolving the synonymy of bacterial taxa, including the correction of deprecated and misspelled names. This deficiency may lead to the unnatural split of some taxa into different entities during the analysis.

It is well known that some fatty acids are highly informative as taxonomic markers, whereas others may vary from strain to strain [49]. Therefore, we have subjected the qualitative fatty acid distribution for each of the 32 newly delineated peak naming windows to a computational analysis using the hierarchical cube operator [9, 26, 33], which performs a roll-up agglomeration on the measured characteristics of the chromatographic peaks according to the standing taxonomic stratification of the bacteria. In essence, this means that all fatty acid profiles are initially grouped according to their identification into the taxonomic hierarchy, and that the agglomerative operation takes place from the most specific levels of the hierarchy towards the most general levels. Table 5.2 shows a small excerpt of the agglomerative roll-up chromatographic peak statistics per taxon for the new fatty

acid peak naming window 11, with an ECL range between 13.800 and 13.826. Complete lists for all investigated new naming windows are included in section B.2.2 of the appendices. The first column of these tables shows the different taxa for which at least one sample was encountered within the FAME database, having a chromatographic peak in the ECL range of the naming window at hand. Indentation was used to reflect the hierarchical relationship of the different taxa. Closely inspecting this data learns that some of the *Brevundimonas diminuta* [35] strains are still encoded using the former name of the species, being *Pseudomonas diminuta*. In a similar way, the identification of some strains of the species *Alteromonas stellipolaris* [46] is clearly misspelled as *Alteromonas stellaepolaris* in the database. Due to this alternative naming, the corresponding samples are erroneously grouped into different taxonomic units for analysis purposes. This hindrance could be overcome by the implementation of a tool for the resolution of known synonyms and misspellings in the taxonomic names. Another improvement of the taxonomic grouping could be realized by the inclusion within the integrated strain database of the information on taxonomic ranks above the genus level. As a result, the hierarchical cube agglomeration would be extended to higher order ranks such as families and orders within the prokaryotes [8].

The following three table columns give an impression on the frequency of detecting a chromatographic peak within the ECL range of the corresponding peak naming window for the taxon at hand. The second column (tot) indicates the total number of fatty acid profiles for the given taxon that are incorporated into the FAME database (recall that the analysis is restricted to the fatty acid profiles of strains from the BCCM™/LMG Bacteria Collection), whereas the first column (`occ`) amounts to the portion of these profiles that shows a chromatographic peak within the relevant ECL range. The fraction of these values, represented as a percentage, is shown in the third column (`rel`). The operation of hierarchical cube agglomeration can be illustrated in a simple manner, by inspection of the second peak frequency column (`tot`). The database contains 23 profiles of strains that are identified as *Photobacterium damselae* subsp. *damselae*, and 3 profiles of *Photobacterium damselae* subsp. *piscicida* strains. Addition of these two values results in a total of 26 strains for the species *Photobacterium damselae*. By adding the number of profiles for all *Photobacterium* species, a total number of 66 profiles for the genus is found. Note however that the total number of fatty acid profiles considered at a certain taxonomic level might be higher than the sum of the number of profiles for all the taxonomic sublevels shown in the peak significance tables. This is for example the case for the genus *Brevundimonas*, for which 99 fatty acid profiles are found within the FAME database. The reason for this shortage of profiles when inspecting the taxonomic sublevels is twofold. In the first place, not all subtaxa of a given taxon necessarily appear in the lists as shown in Table 5.2, either because no samples for these taxa were incorporated in the FAME database, or none of the peak profiles for the taxonomic unit showed a chromatographic peak within the ECL range of the peak naming window at hand. For example in the case of the genus *Brevundimonas* for the peak naming window covering the ECL interval between 13.800 and 13.826, no chromatographic peaks were found for samples of the species *Brevundimonas aurantiaca*, *B. bacteroides* and *B. vesicularis*, while no profiles were present in the database for strains identified as the species *Brevundimonas nasdae*, recently described from strains isolated at the Russian space station Mir [21]. A second group of fatty acid profiles that is not accounted for at the subtaxon level of a given taxon, are the profiles of strains that

						ECL		a^r			
taxon	occ	tot	rel	$\mu(\frac{a_n}{a_t})$	$\mu(\frac{p_n}{p_t})$	μ	σ	μ	σ	$\lfloor\rfloor$	$\lfloor\rfloor$
Actinobacillus sp.	4	4	100.00%	98.12%	84.24%	13.815	0.001	1.30	0.21	1.15	1.60
Actinobacillus equuli	3	3	100.00%	98.19%	84.12%	13.814	0.001	1.20	0.08	1.15	1.30
Actinobacillus equuli subsp. *equuli*	3	3	100.00%	98.19%	84.12%	13.814	0.001	1.20	0.08	1.15	1.30
Actinobacillus lignieresii	1	1	100.00%	97.92%	84.62%	13.815	0.000	1.60	0.00	1.60	1.60
Alteromonas sp.	27	30	90.00%	93.38%	85.97%	13.814	0.001	0.88	0.23	0.41	1.57
Alteromonas macleodii	6	6	100.00%	93.14%	89.34%	13.814	0.001	0.67	0.12	0.52	0.85
Alteromonas stellaepolaris	5	5	100.00%	90.63%	81.39%	13.814	0.001	0.92	0.17	0.72	1.12
Alteromonas stellipolaris	15	15	100.00%	93.29%	83.41%	13.814	0.001	0.98	0.19	0.74	1.57
Aquicella sp.	4	4	100.00%	89.40%	75.66%	13.810	0.001	0.37	0.18	0.22	0.63
Aquicella lusitana	2	2	100.00%	86.31%	75.65%	13.809	0.000	0.50	0.18	0.37	0.63
Aquicella siphonis	2	2	100.00%	92.49%	75.67%	13.810	0.001	0.25	0.04	0.22	0.27
Arcobacter sp.	141	158	89.24%	93.41%	85.91%	13.813	0.001	6.05	4.03	0.40	18.85
Arcobacter butzleri	23	23	100.00%	94.97%	85.61%	13.814	0.001	4.61	1.96	0.85	8.29
Arcobacter cryaerophilus	83	91	91.21%	91.87%	87.57%	13.814	0.001	8.31	3.41	3.04	18.85
Arcobacter skirrowii	35	40	87.50%	95.45%	81.60%	13.812	0.002	1.64	1.51	0.40	6.33
Azospirillum sp.	2	6	33.33%	98.84%	95.96%	13.813	0.001	0.77	0.00	0.77	0.77
Azospirillum irakense	2	2	100.00%	96.51%	87.87%	13.813	0.001	0.77	0.00	0.77	0.77
Brevundimonas sp.	50	99	50.51%	93.81%	82.83%	13.814	0.001	1.11	0.51	0.13	1.87
Brevundimonas alba	2	2	100.00%	89.41%	77.08%	13.817	0.001	0.47	0.09	0.41	0.54
Brevundimonas diminuta	40	51	78.43%	93.40%	81.74%	13.814	0.001	1.32	0.30	0.67	1.87
Brevundimonas intermedia	1	3	33.33%	95.68%	87.66%	13.816	0.000	0.18	0.00	0.18	0.18
Brevundimonas subvibrioides	1	5	20.00%	92.39%	83.40%	13.813	0.000	0.13	0.00	0.13	0.13
Brevundimonas variabilis	2	2	100.00%	90.15%	78.89%	13.815	0.001	0.36	0.13	0.27	0.46
Cellvibrio sp.	24	59	40.68%	97.24%	88.51%	13.815	0.002	0.45	0.19	0.21	0.91
Cellvibrio fibrivorans	9	11	81.82%	97.01%	83.49%	13.816	0.001	0.56	0.23	0.21	0.91
Cellvibrio mixtus	2	2	100.00%	96.68%	82.31%	13.813	0.001	0.48	0.29	0.28	0.69
Cellvibrio mixtus subsp. *mixtus*	2	2	100.00%	96.68%	82.31%	13.813	0.001	0.48	0.29	0.28	0.69
Cellvibrio ostraviensis	5	9	55.56%	97.92%	84.52%	13.814	0.000	0.41	0.11	0.33	0.59
Cellvibrio vulgaris	7	7	100.00%	97.88%	80.10%	13.815	0.001	0.36	0.06	0.31	0.47
Enterovibrio sp.	2	2	100.00%	96.07%	85.99%	13.815	0.000	1.52	0.10	1.45	1.58
Enterovibrio coralii	2	2	100.00%	97.25%	84.38%	13.815	0.000	1.52	0.10	1.45	1.58
Glaciecola sp.	8	9	88.89%	92.45%	86.31%	13.813	0.003	1.67	2.00	0.21	5.02
Glaciecola mesophila	4	5	80.00%	98.45%	96.99%	13.811	0.002	0.26	0.08	0.21	0.38
Glaciecola pallidula	2	2	100.00%	86.38%	80.95%	13.814	0.000	4.82	0.29	4.61	5.02
Glaciecola punicea	2	2	100.00%	89.23%	87.42%	13.815	0.003	1.37	0.15	1.26	1.47
Listonella sp.	51	55	92.73%	97.70%	88.71%	13.814	0.001	0.78	0.32	0.39	1.83
Listonella anguillarum	47	51	92.16%	97.64%	88.61%	13.814	0.001	0.77	0.29	0.52	1.83
Listonella pelagia	4	4	100.00%	98.43%	90.00%	13.814	0.001	0.94	0.61	0.39	1.61
Moritella sp.	5	5	100.00%	84.95%	80.36%	13.815	0.000	7.73	4.60	3.73	13.01
Moritella abyssi	2	2	100.00%	81.80%	84.31%	13.815	0.000	12.75	0.38	12.48	13.01
Moritella marina	1	1	100.00%	88.85%	77.78%	13.815	0.000	3.73	0.00	3.73	3.73
Moritella profunda	2	2	100.00%	86.16%	77.71%	13.815	0.000	4.70	0.15	4.60	4.81
Photobacterium sp.	51	66	77.27%	98.21%	89.11%	13.814	0.001	0.76	0.32	0.21	1.74
Photobacterium angustum	6	7	85.71%	98.99%	90.18%	13.813	0.001	0.43	0.15	0.31	0.73
Photobacterium damselae	21	26	80.77%	97.91%	88.03%	13.814	0.001	0.74	0.27	0.21	1.37
Photobacterium damselae subsp. *damselae*	20	23	86.96%	98.25%	89.87%	13.814	0.001	0.76	0.25	0.46	1.37
Photobacterium damselae subsp. *piscicida*	1	3	33.33%	95.38%	73.94%	13.811	0.000	0.21	0.00	0.21	0.21
Photobacterium eurosenbergii	2	2	100.00%	98.92%	96.23%	13.816	0.001	0.82	0.02	0.81	0.83
Photobacterium leiognathi	19	22	86.36%	98.59%	88.65%	13.814	0.001	0.78	0.27	0.46	1.37
Photobacterium phosphoreum	3	9	33.33%	97.41%	90.93%	13.815	0.001	1.42	0.39	0.99	1.74
Pseudoalteromonas sp.	84	93	90.32%	92.07%	88.40%	13.814	0.001	0.75	0.56	0.13	3.33
Pseudoalteromonas atlantica	6	7	85.71%	96.80%	87.55%	13.814	0.001	1.36	0.19	1.14	1.56
Pseudoalteromonas citrea	2	2	100.00%	92.02%	79.69%	13.813	0.001	0.88	0.06	0.84	0.93
Pseudoalteromonas espeijana	9	9	100.00%	89.71%	88.25%	13.814	0.001	1.22	0.84	0.43	3.14
Pseudoalteromonas flavipulchra	3	4	75.00%	93.71%	88.88%	13.817	0.001	0.18	0.01	0.16	0.19
Pseudoalteromonas haloplanktis	25	28	89.29%	93.77%	91.01%	13.814	0.001	0.62	0.63	0.19	3.33
Pseudoalteromonas issachenkonii	8	8	100.00%	89.30%	85.94%	13.814	0.001	0.54	0.16	0.42	0.92
Pseudoalteromonas luteoviolacea	1	2	50.00%	97.86%	89.85%	13.813	0.000	0.32	0.00	0.32	0.32
Pseudoalteromonas maricaloris	10	12	83.33%	88.37%	85.54%	13.814	0.002	0.32	0.14	0.13	0.52
Pseudoalteromonas nigrifaciens	6	7	85.71%	91.77%	91.16%	13.815	0.001	0.77	0.33	0.42	1.20
Pseudoalteromonas paragorgicola	4	4	100.00%	92.78%	90.08%	13.815	0.001	1.27	0.33	1.03	1.74
Pseudoalteromonas prydzensis	4	4	100.00%	88.66%	81.25%	13.814	0.001	1.06	0.11	0.89	1.12
Pseudoalteromonas ruthenica	6	6	100.00%	92.31%	88.99%	13.814	0.001	0.74	0.21	0.44	1.01
Pseudomonas sp.	259	1024	25.29%	95.22%	87.33%	13.813	0.004	0.59	1.55	0.07	12.57
Pseudomonas aurantiaca	4	4	100.00%	84.62%	57.96%	13.804	0.002	0.41	0.06	0.34	0.47
Pseudomonas beijerinckii	1	1	100.00%	98.68%	84.62%	13.813	0.000	0.55	0.00	0.55	0.55
Pseudomonas diminuta	3	3	100.00%	93.74%	85.51%	13.814	0.001	1.95	0.47	1.62	2.49
Pseudomonas lundensis	4	5	80.00%	62.74%	57.29%	13.816	0.002	5.10	6.09	0.12	12.57
Pseudomonas luteola	3	3	100.00%	99.74%	90.30%	13.813	0.000	0.28	0.07	0.20	0.34
Pseudomonas marginalis	51	80	63.75%	99.22%	89.76%	13.814	0.001	0.28	0.08	0.14	0.60
Pseudomonas marginalis pv. *alfalfae*	10	12	83.33%	99.62%	90.88%	13.814	0.001	0.31	0.07	0.15	0.38
Pseudomonas marginalis pv. *marginalis*	27	41	65.85%	99.26%	90.55%	13.815	0.001	0.28	0.10	0.15	0.60
Pseudomonas marginalis pv. *pastinacae*	5	13	38.46%	99.33%	94.39%	13.814	0.001	0.20	0.04	0.14	0.25
Pseudomonas spinosa	2	2	100.00%	99.00%	88.19%	13.814	0.001	1.00	0.24	0.83	1.17
Shewanella sp.	55	72	76.39%	90.08%	87.04%	13.813	0.001	0.67	0.48	0.08	3.12
Shewanella algae	2	4	50.00%	89.92%	92.17%	13.812	0.001	0.11	0.03	0.08	0.13
Shewanella denitrificans	8	8	100.00%	92.06%	88.47%	13.812	0.000	0.65	0.08	0.52	0.73
Shewanella fidelis	4	4	100.00%	91.49%	89.36%	13.814	0.001	0.35	0.06	0.30	0.43
Shewanella frigidimarina	9	10	90.00%	90.92%	81.90%	13.814	0.001	1.33	0.68	0.95	3.12
Shewanella hanedai	4	4	100.00%	82.46%	84.15%	13.814	0.001	0.67	0.02	0.65	0.70
Shewanella japonica	5	5	100.00%	95.33%	83.93%	13.812	0.001	0.46	0.07	0.36	0.56
Shewanella livingstonensis	3	3	100.00%	88.55%	80.87%	13.813	0.000	1.11	0.13	1.02	1.26
Shewanella marinintestina	3	4	75.00%	87.38%	89.43%	13.813	0.001	0.33	0.02	0.31	0.35
Shewanella olleyana	4	5	80.00%	89.88%	86.04%	13.810	0.001	0.43	0.15	0.31	0.64
Shewanella putrefaciens	3	9	33.33%	89.35%	93.24%	13.813	0.000	0.21	0.05	0.17	0.26
Shewanella sairae	3	3	100.00%	94.85%	85.93%	13.812	0.001	0.62	0.15	0.47	0.76
Shewanella schlegeliana	4	4	100.00%	89.09%	86.80%	13.813	0.001	0.26	0.03	0.23	0.30
Shewanella violacea	3	3	100.00%	85.43%	81.67%	13.813	0.001	1.20	0.20	1.07	1.43

Table 5.2: Small excerpt of the agglomerative roll-up chromatographic peak statistics per taxon for the new fatty acid peak naming window 11, with an ECL range between 13.800 and 13.826.

are only identified at the level of accuracy provided by the taxon itself. For example, in the *Brevundimonas* case, the strains `LMG 9564`, `LMG 9567` and `LMG 11070` are only accurately identified at the genus level.

Furthermore, the rolled up statistics also contain an indication of the peak naming success rate of the TSBA50 peak naming method that is commercially available within the Sherlock MIS. The value $\mu(\frac{a_n}{a_t})$ presents the average (μ) percentage of the cumulative amount of fatty acids within the named peaks (a_n), with respect to the total fatty acid content found within the bacterial cell (a_t). These latter values can be immediately extracted from the Sherlock MIS peak naming report as the named area and the total area respectively, taking into account the reponse factor for correcting the absolute amount of fatty acids as given in formula (5.2). As an alternative measure, the value $\mu(\frac{p_n}{p_t})$ expresses the average percentage of the named peaks (p_n) over the total number of peaks detected in the chromatogram. Only chromatographic peaks within the ECL range [9.000,20.000] are taken into account. Table 5.2 clearly shows that the relative amount of the named peaks can be relatively high for a given taxon, whereas there is still a high fraction of unnamed peaks, and vice versa. These two parameters together thus give an impression whether there is either a large number of low-amount unnamed peaks or rather only a small number of unnamed peaks but with a significant contribution to the total fatty acid content of the relevant taxon, or anything in between.

The peak statistics tables also agglomerate for each taxon the average and standard deviation (σ) of the normalized ECL position, for the chromatographic peaks within the given ECL interval of the corresponding new naming window. Table 5.2 shows little or no variation in the ECL positions of the chromatographic peaks for the new naming window covering the ECL interval between 13.800 and 13.826. This is in complete agreement with the single sharp peak encountered with the corresponding naming window in the histogram presented in section B.1. For naming windows that show one or more shoulders in the histogram, these values may aid in the assignment of the corresponding taxon to one of the multiple spikes.

A final group of values are related to the relative amount of the fatty acids within the newly delineated peak naming window, with respect to the total amount of named fatty acids. Apart from the average and standard deviation, also the mininal ($\lfloor\rfloor$) and maximal ($\lceil\rceil$) relative amounts are shown. It should be noted that the addition of one or more new peak naming windows for the naming of previously unnamed peaks within the chromatographic peak profile, influences the value of the denominator in the expression of the relative area (5.2), thus also the values of all relative amounts calculated by the Sherlock MIS. Because the response factors are missing in the unnamed peaks in the Sherlock MIS sample composition reports, we have applied two simplifications for the calculation of the relative areas. The effect of response factor corrections is ignored by regarding all weights as 1.0 in expression (5.2), and the sum in the denominator is only restricted to all chromatographic peaks in the ECL range between 9.000 and 20.000, instead of a further limitation to the named peaks. Note that as such the value for the denominator is readily available as the total area in the Sherlock MIS sample composition reports, and the resulting simplification assumes that all relevant peaks are named after introduction of the 32 new peak naming

windows. This is true for a majority of the profiles in the current version of our proprietary FAME database.

The significance of the newly discovered peak naming windows for each of the validly described taxonomic units directly follows from the tables presented in appendix B. Although this sheer amount of information at first glance might seem overwhelming or even daunting, it definitely reflects the size and complexity of the knowledge present into our proprietary FAME database. In order to sort out the taxonomic units for which the peaks are most stable, we can apply the same quality threshold that is also used for the elimination of features in the Sherlock MIS library entries that have a very low mean and/or are present in only a small percentage of the samples related to a given taxonomic unit. This quality threshold is based on the product of the average relative amount of the fatty acids per taxon ($\mu(a^r)$) and the fraction of samples which contain a chromatographic peak within the corresponding naming window (rel$/100$). A typical value for the quality threshold suggested by MIDI is 0.25. Application of the quality threshold onto the taxa covered within our proprietary FAME database results in a summary of the taxonomic units for which the newly discovered peaks are highly stable, as is shown in Table 5.3. The first column assigns an identifier to each of the newly delineated naming windows, whereas the second column gives the ECL range that is covered by the naming window. The `occ` column indicates the total number of fatty acid profiles in our proprietary FAME database for which a peak was detected in the corresponding naming window. This value hence is not restricted to the profiles associated to a strain contained within the BCCM™/LMG Bacteria Collection, but takes into account all profiles, regardless of the quality or precision of their taxonomic identification. The last column enumerates all taxonomic units for which the quality threshold was at least 0.25. For reasons of compactness, the taxa mentioned in this list were restricted to the lowest level in the hierarchy for which the fatty acid peak was found and the few samples of anaerobic species incorporated into our proprietary FAME database were discarded. Also, if two or more species or subspecies of the same genus appear in the list, the genus name is only fully given with the first scientific name, and abbreviated to the first letter of the name in all subsequent scientific names. The value between brackets represents the average percentage of the relative fatty acid amount found in the samples of the taxonomic unit at hand. In order to highlight the fact that a given naming window is highly specific for a certain genus, the species and subspecies of that genus are printed in bold face in Table 5.2 whenever a chromatographic peak was found in more than two thirds of the species for the given genus.

Comparison of the peak frequencies calculated for the new peak naming windows with those of the peaks named with the Sherlock MIS TSBA50 peak naming method learns that the newly discovered peaks only occur in a moderate to low number of the profiles in our proprietary FAME database. Not very surprising, because we previously observed in subsection 5.3.3 that 83% of the peaks were already named by the TSBA50 peak naming method. Nonetheless, peak 11, covering the ECL range between 13.800 and 13.826, was found in 21.4% of the fatty acid profiles in our proprietary FAME database. Moss & Lambert-Fair have identified this fatty acid peak as 14:1 ω7c by means of mass spectrometry [23], whereas chromatographic peaks on the same position are interpreted as 13:0 iso 2OH according to the peak naming table of the Culture Collection at the University

ID	ECL	occ	taxa for which the histogram peak is significantly found
1	9.201 9.246	2253	*Acidomonas methanolica* (0.28%), *Azorhizobium caulinodans* (4.44%), *Bacillus carotarum* (0.76%), *B. cohnii* (2.15%), *B. flexus* (1.70%), *B. halodurans* (0.72%), *B. macroides* (0.70%), *B. shacketonii* (0.94%), *B. similibadius* (0.93%), *Blastobacter denitrificans* (0.36%), *Brachybacterium alimentarium* (0.45%), *B. nesterenkovii* (1.58%), *B. tyrofermentans* (0.73%), *Enterococcus saccharolyticus* (0.37%), *Erwinia tracheiphila* (1.72%), *Kurthia zopfii* (0.67%), *Muricauda ruestringensis* (1.24%), *Paenibacillus alginolyticus* (0.93%), *Paralactobacillus selangorensis* (0.72%), *Pseudomonas beijerinckii* (0.77%), *Roseobacter denitrificans* (0.43%), *R. litoralis* (0.44%), *Ruegeria algicola* (0.36%), *Staleya guttiformis* (0.55%), *Streptomyces roseoviolaceus* (0.27%), *Sulfitobacter mediterraneus* (0.29%), *S. pontiacus* (0.31%), *Ureibacillus terrenus* (1.09%), *Vibrio pomeroyi* (1.88%)
2	10.067 10.099	2903	*Aeromonas allosaccharophila* (0.36%), *A. culicicola* (0.48%), *A. enteropelogenes* (0.36%), *Enterovibrio coralii* (0.76%), *Listonella anguillarum* (1.17%), *Rhodobacter sphaeroides* (0.32%), *Vibrio mediterranei* (0.47%), *Yersinia enterocolitica* subsp. *enterocolitica* (2.99%)
3	10.464 10.487	1828	*Acidomonas methanolica* (1.63%), *Blastobacter denitrificans* (0.92%), ***Cellvibrio fibrivorans* (0.78%)**, ***C. fulvus* (0.87%)**, ***C. mixtus* subsp. *mixtus* (0.90%)**, ***C. ostraviensis* (0.72%)**, ***C. vulgaris* (0.66%)**, *Collimonas fungivorans* (0.38%), *Pseudomonas aurantiaca* (2.28%), *P. chlororaphis* (0.76%), *P. corrugata* (0.52%), *P. frederiksbergensis* (1.50%), *P. putida* (0.68%), *P. umsongensis* (0.58%), *Ruegeria algicola* (0.78%)
4	11.187 11.206	901	*Acidomonas methanolica* (0.44%), *Bacillus vedderi* (1.51%), *Blastobacter denitrificans* (0.48%), *Cellvibrio fulvus* (0.27%), *C. mixtus* subsp. *mixtus* (0.31%), *C. vulgaris* (0.27%), *Pseudomonas aurantiaca* (0.80%), *P. frederiksbergensis* (0.45%)
5	11.960 11.976	1788	*Blastobacter denitrificans* (0.63%), *Bradyrhizobium elkanii* (0.49%), *B. liaoningense* (0.94%), *Ochrobactrum tritici* (0.75%), *Pseudomonas alcaligenes* (5.04%), *P. aurantiaca* (1.58%), *P. beteli* (0.31%), *P. hibiscicola* (0.29%), *P. stutzeri* (7.98%), *Psychrobacter fozii* (0.73%), *P. glacincola* (0.70%), *Rhodovulum sulfidophilum* (0.36%), *Stenotrophomonas africana* (0.27%), *S. maltophilia* (0.32%), *S. rhizophila* (0.28%), *Xanthomonas arboricola* pv. *celebensis* (0.35%), *X. arboricola* pv. *fragariae* (0.37%), *X. arboricola* pv. *poinsettiicola* (0.27%), *X. axonopodis* pv. *coracanae* (0.35%), *X. axonopodis* pv. *desmodii* (0.37%), *X. axonopodis* pv. *lespedezae* (0.29%), *X. axonopodis* pv. *martyniicola* (0.33%), *X.* sp. pv. *cannabis* (0.32%), *X.* sp. pv. *cannae* (0.42%), *X.* sp. pv. *coriandri* (0.38%), *X.* sp. pv. *lantanae* (0.34%), *X.* sp. pv. *nigromaculans* (0.33%), *X.* sp. pv. *phormiicola* (0.31%), *X.* sp. pv. *vitiscarnosae* (0.28%), *X.* sp. pv. *zantedeschiae* (0.45%), *X. sacchari* (0.29%), *X. theicola* (0.36%), *X. translucens* pv. *phlei* (0.28%)
6	12.340 12.358	709	*Brevibacillus centrosporus* (5.70%), *Pseudomonas aurantiaca* (1.68%), *P. chlororaphis* (0.73%), *P. frederiksbergensis* (3.14%), *P. umsongensis* (0.56%)
7	12.770 12.787	796	*Loktanella salsilacus* (0.88%), ***Pseudoalteromonas citrea* (0.39%)**, ***P. espeijana* (1.12%)**, ***P. flavipulchra* (0.56%)**, ***P. haloplanktis* (0.99%)**, ***P. issachenkonii* (1.11%)**, ***P. maricaloris* (1.06%)**, ***P. nigrifaciens* (1.61%)**, ***P. paragorgicola* (0.85%)**, ***P. prydzensis* (1.21%)**, ***P. ruthenica* (0.38%)**, *Rheinheimera baltica* (1.05%), *Roseobacter gallaeciensis* (0.71%)
8	13.136 13.149	341	***Aquicella lusitana* (0.79%)**, ***A. siphonis* (0.43%)**, *Pseudomonas frederiksbergensis* (0.75%), *Shewanella denitrificans* (0.67%), *S. frigidimarina* (0.29%), *S. hanedai* (0.43%), *S. japonica* (0.62%), *S. livingstonensis* (0.27%), *S. olleyana* (0.64%), *S. sairae* (0.55%), *S. violacea* (0.72%)
9	13.515 13.529	840	*Bacillus firmus* (0.92%), *Bifidobacterium adolescentis* (1.13%), *B. cuniculi* (1.33%), *Brevibacterium epidermidis* (1.51%), *Enterococcus pseudoavium* (2.22%), *Gulosibacter molinativorax* (19.63%), *Microscilla arenaria* (0.51%), *Moraxella* subgen. *Moraxella bovis* (1.58%), *M.* subgen. *Moraxella nonliquefaciens* (1.79%), ***Moritella marina* (0.77%)**, ***M. profunda* (0.46%)**, *Mycobacterium aurum* (6.92%), *M. gilvum* (5.02%), *M. vanbaalenii* (7.26%), *Pseudomonas alcaligenes* (4.09%), *Streptomyces canescens* (1.34%), *S. citreus* (1.60%), *S. coelicolor* (2.94%), *S. felleus* (1.07%), *S. limosus* (0.73%), *S. odorifer* (1.77%), *S. paucisporogenes* (0.34%), *S. rutgersensis* subsp. *rutgersensis* (2.38%), *S. tendae* (0.91%), *Vibrio logei* (0.79%), *V. superstes* (0.42%), *Xanthomonas axonopodis* pv. *dieffenbachiae* (5.40%)

Table 5.3: New peak naming windows derived from the peak occurrence histogram, with an overview of the taxa for which the corresponding fatty acid peaks are significant according to a quality threshold of 0.25. The first column assigns an identifier to each of the newly delineated naming windows, whereas the second column gives the ECL range that is covered by the naming window. The occ column indicates the total number of fatty acid profiles in our proprietary FAME database for which a peak was detected in the corresponding naming window. The values between brackets after the scientific name of the taxa wherefore the peak was found to be significant indicates the average percentage of the relative fatty acid amount found in the samples of the taxonomic unit at hand, and the species and subspecies for a given genus are printed in bold face whenever a chromatographic peak was found in more than two thirds of the species for the given genus.

ID	ECL	occ	taxa for which the histogram peak is significantly found
10	13.761 13.772	703	*Acetobacter orleanensis* (1.31%), *Aequorivita crocea* (0.27%), *Aerococcus viridans* (1.27%), *Arcanobacterium pyogenes* (0.35%), *Bacillus azotoformans* (0.90%), *Bifidobacterium animalis* (1.06%), *B. bifidum* (0.88%), *B. catenulatum* (1.37%), *B. infantis* (0.89%), *Corynebacterium afermentans* subsp. *afermentans* (0.28%), *C. argentoratense* (0.29%), *C. coyleae* (0.32%), *C. flavescens* (0.65%), *C. imitans* (0.37%), *C. pseudotuberculosis* (0.39%), *C. seminale* (0.57%), *Macrococcus brunensis* (0.94%), *M. hajekii* (1.68%), *M. lamae* (1.08%), *Marinobacter hydrocarbonoclasticus* (0.96%), *Mycobacterium aichiense* (0.37%), *M. peregrinum* (0.38%), *Oleiphilus messinensis* (0.51%), *Paenibacillus antarcticus* (0.78%), *Thermomonas haemolytica* (0.79%), *Vagococcus fluvialis* (1.51%), *Vibrio kanaloaei* (0.58%), *Xanthomonas populi* (1.59%)
11	13.800 13.826	10484	*Actinobacillus equuli* subsp. *equuli* (1.20%), *A. lignieresii* (1.60%), *Aeromonas culicicola* (0.39%), *A. enteropelogenes* (0.35%), *A. hydrophila* subsp. *dhakensis* (0.37%), *A. ichthiosmia* (0.49%), *A. popoffii* (0.41%), *A. salmonicida* subsp. *masoucida* (0.36%), *A. schubertii* (0.63%), *Agrobacterium luteum* (3.97%), ***Alteromonas macleodii* (0.67%), *A. stellaepolaris* (0.92%), *A. stellipolaris* (0.98%)**, *Aquaspirillum polymorphum* (1.12%), *Aquicella lusitana* (0.50%), ***Arcobacter butzleri* (4.61%), *A. cryaerophilus* (8.31%), *A. skirrowii* (1.64%)**, *Azospirillum irakense* (0.77%), *Brenneria paradisiaca* (0.66%), *Brevundimonas alba* (0.47%), *B. diminuta* (1.32%), *B. variabilis* (0.36%), *Cellvibrio fibrivorans* (0.56%), *C. mixtus* subsp. *mixtus* (0.48%), *C. vulgaris* (0.36%), *Devosia neptuniae* (1.21%), ***Enterococcus avium* (2.88%), *E. canis* (1.56%), *E. casseliflavus* (1.40%), *E. columbae* (5.07%), *E. dispar* (3.46%), *E. durans* (1.26%), *E. faecalis* (1.51%), *E. faecium* (2.62%), *E. flavescens* (0.93%), *E. gallinarum* (1.46%), *E. gilvus* (1.41%), *E. hirae* (1.38%), *E. malodoratus* (3.67%), *E. mundtii* (1.28%), *E. pseudoavium* (2.17%), *E. raffinosus* (2.64%)**, *Enterovibrio coralii* (1.52%), *Flavobacterium gillisiae* (0.52%), *Flexibacter aurantiacus* subsp. *copepodarum* (0.29%), *Frateuria aurantia* (0.43%), *Fusobacterium nucleatum* subsp. *nucleatum* (0.38%), ***Glaciecola pallidula* (4.82%), *G. punicea* (1.37%)**, *Grimontia hollisae* (1.98%), *Halomonas cupida* (0.52%), *H. pacifica* (1.76%), *H. venusta* (0.45%), *Lactobacillus brevis* (1.02%), *L. delbrueckii* subsp. *bulgaricus* (0.81%), ***Lactococcus garvieae* (2.35%), *L. lactis* subsp. *lactis* (0.91%)**, *Leuconostoc mesenteroides* subsp. *dextranicum* (1.28%), ***Listonella anguillarum* (0.77%), *L. pelagia* (0.94%)**, *Marinospirillum minutulum* (7.34%), *Microbacterium terregens* (1.31%), ***Moritella abyssi* (12.75%), *M. marina* (3.73%), *M. profunda* (4.70%)**, *Mycoplana bullata* (2.05%), *Neisseria flavescens* (0.70%), *N. perflava* (0.88%), *Oceanimonas doudoroffii* (0.67%), *Oleiphilus messinensis* (0.42%), *Pectobacterium cacticida* (0.42%), *P. wasabiae* (0.27%), ***Photobacterium angustum* (0.43%), *P. damselae* subsp. *damselae* (0.76%), *P. eurosenbergii* (0.82%), *P. leiognathi* (0.78%), *P. phosphoreum* (1.42%)**, *Plesiomonas shigelloides* (0.58%), ***Pseudoalteromonas atlantica* (1.36%), *P. citrea* (0.88%), *P. espeijana* (1.22%), *P. haloplanktis* (0.62%), *P. issachenkonii* (0.54%), *P. maricaloris* (0.32%), *P. nigrifaciens* (0.77%), *P. paragorgicola* (1.27%), *P. prydzensis* (1.06%), *P. ruthenica* (0.74%)**, *Pseudomonas aurantiaca* (0.41%), *P. beijerinckii* (0.55%), *P. diminuta* (1.95%), *P. lundensis* (5.10%), *P. luteola* (0.28%), *P. marginalis* pv. *alfalfae* (0.31%), *P. spinosa* (1.00%), *Psychromonas profunda* (12.45%), *Rheinheimera baltica* (2.15%), *Roseobacter gallaeciensis* (0.84%), *R. litoralis* (0.61%), *Salinivibrio costicola* subsp. *costicola* (1.23%), ***Shewanella denitrificans* (0.65%), *S. fidelis* (0.35%), *S. frigidimarina* (1.33%), *S. hanedai* (0.67%), *S. japonica* (0.46%), *S. livingstonensis* (1.11%), *S. olleyana* (0.43%), *S. sairae* (0.62%), *S. schlegeliana* (0.26%), *S. violacea* (1.20%)**, *Sphaerotilus natans* (1.36%), *Streptococcus thoraltensis* (1.27%), ***Vibrio aestuarianus* (0.55%), *V. anguillarum* (0.90%), *V. brasiliensis* (0.66%), *V. campbellii* (0.36%), *V. chagasii* (0.62%), *V. cincinnatiensis* (0.26%), *V. coralliilyticus* (0.35%), *V. crassostreae* (0.55%), *V. cyclitrophicus* (0.53%), *V. diabolicus* (0.26%), *V. diazotrophicus* (0.52%), *V. ezurae* (1.76%), *V. fischeri* (0.88%), *V. fortis* (0.61%), *V. gallicus* (0.75%), *V. halioticoli* (1.10%), *V. harveyi* (0.36%), *V. hepatarius* (0.57%), *V. hispanicus* (0.51%), *V. kanaloae* (0.49%), *V. kanaloaei* (0.67%), *V. lentus* (1.12%), *V. logei* (1.88%), *V. mediterranei* (0.60%), *V. metschnikovii* (0.52%), *V. mytili* (0.56%), *V. navarrensis* (0.36%), *V. neonatus* (1.21%), *V. neptunius* (0.43%), *V. nereis* (0.66%), *V. nigripulchritudo* (0.53%), *V. ordalii* (1.14%), *V. orientalis* (0.90%), *V. pacinii* (0.61%), *V. parahaemolyticus* (0.31%), *V. pelagius* (0.75%), *V. pomeroyi* (0.74%), *V. proteolyticus* (0.31%), *V. rotiferianus* (0.39%), *V. shilonii* (0.38%), *V. splendidus* (0.75%), *V. tasmaniensis* (1.42%), *V. tubiashii* (0.66%), *V. vulnificus* (0.39%), *V. xuii* (0.43%)**, *Xanthomonas arboricola* pv. *corylina* (2.88%), *X. hortorum* pv. *taraxaci* (0.26%), *X.* sp. pv. *gummisudans* (0.33%), *X. translucens* pv. *phlei* (0.29%), *Zymomonas mobilis* subsp. *mobilis* (2.19%)
12	14.570 14.581	804	***Brachybacterium alimentarium* (2.42%), *B. conglomeratum* (1.66%), *B. faecium* (0.91%), *B. fresconis* (1.25%), *B. paraconglomeratum* (1.07%), *B. rhamnosum* (2.12%), *B. sacelli* (1.38%), *B. tyrofermentans* (2.38%)**, *Plantibacter flavus* (0.59%)
13	14.811 14.817	288	*Arenibacter latericius* (0.62%), *Brumimicrobium glaciale* (0.75%), *Cellulophaga algicola* (1.21%), *C. baltica* (0.63%), *Cytophaga latercula* (0.59%), *Flavobacterium aquatile* (0.88%), *F. columnare* (0.90%), *F. degerlachei* (0.98%), *F. frigoris* (0.55%), *F. fryxellicola* (0.72%), *F. gelidilacus* (1.00%), *F. hibernum* (1.34%), *F. micromati* (1.31%), *F. aurantiacus* subsp. *excathedrus* (1.93%), *F. tractuosus* (1.25%), *Muricauda ruestringensis* (0.60%)

Table 5.4: Continuation of Table 5.3

ID	ECL	occ	taxa for which the histogram peak is significantly found
14	15.160 15.193	3366	***Actinobacillus equuli* subsp. *equuli* (0.39%), *A. lignieresii* (0.48%)**, *Aeromonas hydrophila* subsp. *dhakensis* (0.35%), *Bacillus horikoshii* (0.85%), *Brachybacterium frigoritolerans* (2.15%), *B. nesterenkovii* (2.01%), *Brenneria paradisiaca* (0.43%), *Burkholderia glumae* (1.19%), *Corynebacterium casei* (1.00%), *C. singulare* (1.26%), *Curtobacterium plantarum* (0.42%), *Erwinia billingiae* (0.39%), *E. persicina* (0.71%), *E. rhapontici* (0.56%), *Flexibacter ruber* (1.67%), *Hafnia alvei* (0.50%), *Klebsiella pneumoniae* subsp. *ozaenae* (0.33%), *Pantoea agglomerans* (0.34%), ***Pectobacterium cacticida* (0.34%), *P. carotovorum* subsp. *odoriferum* (0.38%), *P. chrysanthemi* (0.45%), *P. wasabiae* (0.34%)**, *Polaribacter glomeratus* (2.44%), *Proteus vulgaris* (0.40%), *Pseudomonas corrugata* (1.26%), *P. tremae* (0.40%), *Rhodobacter sphaeroides* (0.32%), *Sporosarcina pasteurii* (0.71%), *Tenacibaculum maritimum* (0.57%), *Xanthomonas hyacinthi* (1.93%)
15	15.266 15.282	5359	*Acidovorax avenae* subsp. *avenae* (0.53%), *Aeromonas ichthiosmia* (0.28%), *Alteromonas macleodii* (0.40%), *Arcobacter nitrofigilis* (1.06%), *A. skirrowii* (2.90%), *Brevundimonas alba* (1.77%), *Devosia neptuniae* (0.88%), *Glaciecola pallidula* (5.24%), *G. punicea* (5.73%), *Grimontia hollisae* (0.51%), *Ketogulonicigenium robustum* (0.54%), ***Listonella anguillarum* (1.42%), *L. pelagia* (0.87%)**, *Loktanella fryxellensis* (2.21%), *Marinospirillum minutulum* (6.79%), ***Moritella abyssi* (3.46%), *M. marina* (2.90%), *M. profunda* (5.35%)**, *Neisseria flavescens* (0.69%), *Paracoccus denitrificans* (1.25%), *P. zeaxanthinifaciens* (1.39%), *Pseudoalteromonas citrea* (2.50%), *P. prydzensis* (1.39%), *Pseudomonas aurantiaca* (2.17%), *P. chlororaphis* (0.95%), *P. frederiksbergensis* (0.86%), *P. umsongensis* (0.94%), *Rheinheimera baltica* (5.07%), ***Roseobacter denitrificans* (5.67%), *R. litoralis* (4.66%)**, *Ruegeria algicola* (0.90%), *Shewanella frigidimarina* (1.64%), *S. livingstonensis* (1.28%), *Staleya guttiformis* (3.92%), *Streptomyces antibioticus* (2.01%), *S. caelestis* (1.51%), *S. netropsis* (1.41%), *Sulfitobacter brevis* (2.81%), *S. mediterraneus* (3.82%), *S. pontiacus* (3.09%), ***Vibrio aestuarianus* (0.97%), *V. anguillarum* (1.43%), *V. brasiliensis* (0.64%), *V. chagasii* (0.72%), *V. cincinnatiensis* (0.28%), *V. coralliilyticus* (0.82%), *V. crassostreae* (1.66%), *V. cyclitrophicus* (1.20%), *V. diazotrophicus* (1.04%), *V. ezurae* (0.40%), *V. fischeri* (1.46%), *V. fortis* (0.67%), *V. hepatarius* (0.95%), *V. hispanicus* (1.05%), *V. kanaloae* (0.65%), *V. kanaloaei* (1.09%), *V. lentus* (2.18%), *V. logei* (1.01%), *V. metschnikovii* (0.66%), *V. navarrensis* (0.29%), *V. neptunius* (0.68%), *V. nereis* (0.93%), *V. nigripulchritudo* (2.33%), *V. ordalii* (1.51%), *V. orientalis* (0.78%), *V. pacinii* (1.25%), *V. pelagius* (0.79%), *V. pomeroyi* (1.57%), *V. splendidus* (1.59%), *V. tasmaniensis* (1.37%), *V. tubiashii* (0.98%), *V. vulnificus* (0.36%)**
16	15.403 15.423	317	*Aeromonas jandaei* (2.83%), *A. trota* (1.17%), *A. veronii* biogrp. *sobria* (1.66%), *Flavobacterium tirrenicum* (4.69%), *Flexibacter elegans* (0.54%), *Pseudomonas pictorum* (0.67%), *Stenotrophomonas nitritireducens* (0.26%), *Streptomyces argenteolus* (17.79%), *S. aureofaciens* (14.36%), *S. bluensis* (11.99%), *S. cinnamoneus* (26.44%), *S. cinnamoneus* subsp. *cinnamoneus* (26.44%), *S. coriofaciens* (25.09%), *S. kentuckensis* (24.03%), *S. sampsonii* (30.50%), *Xanthomonas sacchari* (0.29%)
17	15.935 15.955	830	*Aequorivita lipolytica* (0.92%), *Algoriphagus ratkowskyi* (3.45%), *Arenibacter latericius* (1.30%), *Arthrobacter aurescens* (6.12%), *Capnocytophaga granulosa* (0.51%), *C. haemolytica* (0.56%), ***Cellulophaga baltica* (1.78%), *C. fucicola* (0.90%), *C. lytica* (0.72%), *C. pacifica* (0.88%)**, *Chitinophaga pinensis* (35.88%), *Chryseobacterium indoltheticum* (0.56%), *C. michiganensis* subsp. *insidiosus* (10.85%), *Cytophaga aurantiaca* (45.97%), *C. hutchinsonii* (38.63%), *C. marinoflava* (0.40%), *Flavobacterium hydatis* (0.67%), *F. johnsoniae* (0.70%), *F. succinicans* (0.76%), *F. aurantiacus* subsp. *copepodarum* (3.77%), *F. elegans* (19.42%), *Gillisia limnaea* (4.40%), *Leeuwenhoekiella aequorea* (0.78%), *Microscilla furvescens* (28.01%), *Myroides odoratimimus* (0.50%), *Nocardioides simplex* (3.63%), *Paenibacillus ehimensis* (0.72%), *P. larvae* subsp. *larvae* (2.17%), *P. larvae* subsp. *pulvifaciens* (2.46%), *Salegentibacter salegens* (3.65%), *Sphingobacterium multivorum* (1.19%), *Streptomyces albofaciens* (14.85%), *S. anandii* (7.43%), *S. flavofungini* (0.89%), *S. peruviensis* (0.82%), ***Tenacibaculum maritimum* (2.31%), *T. ovolyticum* (1.06%)**, *Xanthomonas arboricola* pv. *corylina* (2.46%), *X. axonopodis* pv. *malvacearum* (15.47%)
18	16.076 16.099	2286	***Achromobacter insolitus* (0.68%), *A. piechaudii* (0.86%), *A. ruhlandii* (0.67%), *A. spanius* (0.75%), *A. xylosoxidans* subsp. *denitrificans* (0.65%), *A. xylosoxidans* subsp. *xylosoxidans* (0.49%)**, *Aeromonas bestiarum* (32.97%), *A. veronii* biogrp. *sobria* (32.06%), *Alcaligenes faecalis* (0.75%), *Aquaspirillum autotrophicum* (0.97%), *Arthrobacter oxydans* (0.82%), *A. polychromogenes* (0.86%), ***Bordetella avium* (1.15%), *B. bronchiseptica* (0.79%), *B. hinzii* (1.16%), *B. holmesii* (0.84%), *B. parapertussis* (1.05%), *B. trematum* (0.90%)**, *Burkholderia plantarii* (0.67%), *Caenibacterium thermophilum* (0.80%), *Flexibacter roseolus* (2.69%), *Gardnerella vaginalis* (4.21%), *Herbaspirillum lusitanum* (0.36%), *Kerstersia gyiorum* (1.18%), *Klebsiella oxytoca* (0.71%), *Nocardioides simplex* (1.05%), *Oxalicibacterium flavum* (1.51%), *Paenibacillus validus* (5.72%), *Pandoraea norimbergensis* (0.51%), *Pedobacter piscium* (3.54%), *Photobacterium damselae* subsp. *piscicida* (0.99%), *Pigmentiphaga kullae* (1.07%), *Pseudomonas stutzeri* (14.37%), *Ralstonia taiwanensis* (0.55%), *Schlegelella thermodepolymerans* (0.76%), *Streptococcus pyogenes* (0.75%), *Streptomyces antibioticus* (3.35%), *S. caelestis* (6.83%), *S. netropsis* (1.40%), *Taxeobacter chitinovorans* (0.79%), *Thermoanaerobacter thermohydrosulfuricus* (1.15%), *Yersinia enterocolitica* subsp. *enterocolitica* (0.59%)

Table 5.5: Continuation of Table 5.3

ID	ECL	occ	taxa for which the histogram peak is significantly found
19	16.100 16.120	517	*Aeromonas bestiarum* (21.17%), *A. caviae* (38.37%), *A. encheleia* (5.24%), *A. eucrenophila* (7.24%), *A. hydrophila* subsp. *hydrophila* (39.09%), *A. media* (41.10%), ***Brachybacterium alimentarium* (12.78%), *B. conglomeratum* (9.81%), *B. faecium* (2.83%), *B. fresconis* (3.91%), *B. paraconglomeratum* (6.00%), *B. rhamnosum* (10.15%), *B. sacelli* (3.80%), *B. tyrofermentans* (10.34%)**, *Burkholderia cepacia* (6.96%), *Enterococcus avium* (12.17%), *E. casseliflavus* (5.81%), *E. cecorum* (5.45%), *E. columbae* (2.01%), *E. durans* (13.86%), *E. faecium* (19.15%), *E. gallinarum* (10.75%), *E. mundtii* (22.62%), *E. sulfureus* (7.24%), *E. villorum* (17.53%), *Flexibacter ruber* (14.74%), *Nocardioides albus* (0.88%), *Plantibacter flavus* (2.85%), *Pseudomonas aeruginosa* (19.41%), *P. lundensis* (27.09%), *P. putida* (23.03%), *Rhizobium leguminosarum* (6.75%), *Streptococcus uberis* (1.97%), *Streptomyces murinus* (0.88%), *S. rimosus* subsp. *rimosus* (0.86%), *Subtercola pratensis* (0.37%), *Xanthomonas arboricola* pv. *corylina* (15.18%)
20	16.193 16.200	403	*Bacillus halmapalus* (0.60%), *B. horikoshii* (1.27%), *B. pseudalcaliphilus* (1.22%), *Flexibacter ruber* (3.93%), *Pseudomonas aeruginosa* (21.83%), *Psychrobacter immobilis* (0.36%), *Thermoanaerobacter thermohydrosulfuricus* (0.99%), *Xanthomonas albilineans* (0.81%), *X. hyacinthi* (1.07%)
21	17.306 17.325	1280	*Aeromonas popoffii* (0.33%), *Alteromonas stellaepolaris* (0.52%), *A. stellipolaris* (0.59%), *Flavobacterium flevense* (0.47%), *F. fryxellicola* (0.38%), *F. psychrolimnae* (0.37%), *Glaciecola pallidula* (0.34%), *Loktanella salsilacus* (3.57%), *Pseudomonas amygdali* (5.54%), *P. aurantiaca* (0.61%), *P. caricapapayae* (0.52%), *P. coronafaciens* (0.64%), *P. syringae* pv. *oryzae* (1.12%), *P. syringae* pv. *pisi* (1.11%), *P. syringae* pv. *tagetis* (0.60%), *Psychromonas profunda* (2.51%), *Shewanella sairae* (0.39%), *Tenacibaculum ovolyticum* (0.88%), *Vibrio algosus* (0.42%), *V. kanaloaei* (1.49%), *V. ordalii* (0.29%), *Xanthomonas* sp. pv. *gummisudans* (0.51%), *X. translucens* pv. *hordei* (0.81%), *X. translucens* pv. *phlei* (0.78%), *X. translucens* pv. *poae* (0.62%), *X. translucens* pv. *undulosa* (0.66%)
22	17.588 17.600	146	*Chryseobacterium scophthalmum* (0.86%), *Flammeovirga aprica* (0.95%), *Microscilla arenaria* (0.91%), *Streptomyces albidus* (0.42%), *S. alboviridis* (0.51%), *S. anandii* (0.94%), *S. anulatus* (0.47%), *S. aureofaciens* (3.20%), *S. bacillaris* (0.85%), *S. caeruleus* (1.94%), *S. canescens* (0.86%), *S. citreus* (0.66%), *S. curacoi* (0.37%), *S. echinatus* (1.54%), *S. fluorescens* (0.69%), *S. griseocarneus* (1.32%), *S. kentuckensis* (2.58%), *S. limosus* (1.20%), *S. murinus* (1.14%), *S. nogalater* (1.83%), *S. noursei* (2.95%), *S. oligocarbophilus* (0.42%), *S. pluricolorescens* (2.19%), *S. rimosus* (1.29%), *S. rimosus* subsp. *rimosus* (1.29%), *S. rutgersensis* (0.46%), *S. rutgersensis* subsp. *rutgersensis* (0.46%), *S. septatus* (1.03%), *S. viridifaciens* (0.36%), *Tenacibaculum ovolyticum* (2.42%)
23	17.600 17.616	279	*Ahrensia kielensis* (2.33%), *Cryomorpha ignava* (1.96%), ***Loktanella fryxellensis* (8.11%), *L. salsilacus* (5.86%), *L. vestfoldensis* (4.04%)**, *Roseobacter denitrificans* (9.66%), *R. litoralis* (14.10%), *Ruegeria algicola* (6.54%), *Staleya guttiformis* (11.98%), *Streptomyces tendae* (0.90%), ***Sulfitobacter brevis* (7.47%), *S. delicatus* (1.69%), *S. dubius* (1.05%), *S. mediterraneus* (14.40%)**, ***Tenacibaculum maritimum* (7.94%), *T. ovolyticum* (2.18%)**
24	17.743 17.757	3155	***Achromobacter insolitus* (0.68%), *A. piechaudii* (0.78%), *A. ruhlandii* (0.60%), *A. spanius* (0.62%), *A. xylosoxidans* subsp. *denitrificans* (0.62%), *A. xylosoxidans* subsp. *xylosoxidans* (0.50%)**, *Alcaligenes faecalis* (0.61%), *Aquaspirillum autotrophicum* (0.93%), *Bacillus drentensis* (7.24%), *B. thermacidophilum* subsp. *porcinum* (0.83%), ***Bordetella avium* (1.14%), *B. bronchiseptica* (0.94%), *B. holmesii* (0.79%), *B. parapertussis* (1.60%), *B. trematum* (0.85%)**, *Burkholderia ambifaria* (0.42%), *B. multivorans* (0.49%), *B. plantarii* (0.60%), *Caenibacterium thermophilum* (0.66%), *Hafnia alvei* (0.39%), *Halomonas marina* (0.31%), *Herbaspirillum lusitanum* (0.33%), *Kerstersia gyiorum* (1.11%), ***Klebsiella oxytoca* (0.62%), *K. pneumoniae* subsp. *pneumoniae* (0.67%)**, *Lactobacillus plantarum* (8.99%), *Oxalicibacterium flavum* (1.16%), *Pandoraea apista* (0.38%), *P. norimbergensis* (0.36%), *Photobacterium damselae* subsp. *piscicida* (1.07%), *Pigmentiphaga kullae* (1.00%), *Pseudomonas aurantiaca* (0.26%), *Rahnella aquatilis* (0.53%), *Ralstonia taiwanensis* (0.50%), *Schlegelella thermodepolymerans* (0.60%), *Streptococcus pluranimalium* (17.67%), *S. pyogenes* (1.28%), *Vagococcus fluvialis* (39.69%), *Yersinia enterocolitica* subsp. *enterocolitica* (0.49%)
25	17.783 17.809	1382	*Aeromonas encheleia* (7.57%), *A. eucrenophila* (6.30%), *A. veronii* biogrp. *sobria* (11.36%), *Belliella baltica* (0.43%), *Capnocytophaga gingivalis* (0.50%), *C. ochracea* (0.44%), *C. sputigena* (0.41%), ***Chryseobacterium balustinum* (0.56%), *C. gleum* (0.54%), *C. indologenes* (0.52%), *C. indoltheticum* (0.51%), *C. joostei* (0.41%), *C. meningosepticum* (0.47%)**, *Cytophaga aurantiaca* (0.73%), *C. latercula* (0.31%), *C. marinoflava* (0.38%), *Empedobacter brevis* (0.57%), *Enterococcus avium* (24.49%), *E. dispar* (2.64%), *E. faecium* (33.54%), *E. flavescens* (49.05%), *E. gilvus* (18.60%), *E. pseudoavium* (17.26%), *E. raffinosus* (21.80%), *Flavobacterium aquatile* (0.39%), *F. frigidarium* (0.47%), *F. hydatis* (0.32%), *F. johnsoniae* (0.40%), *F. limicola* (0.32%), *F. pectinovorum* (0.57%), *Flexibacter aurantiacus* subsp. *excathedrus* (0.30%), *F. ruber* (0.38%), *Lactobacillus plantarum* (39.65%), *Lactococcus lactis* subsp. *lactis* (11.20%), ***Myroides odoratimimus* (0.37%), *M. odoratus* (0.45%)**, *Pedobacter heparinus* (0.53%), *P. saltans* (0.63%), *Promyxobacterium flavum* (0.58%), *Pseudomonas aeruginosa* (41.07%), *P. alcaligenes* (36.32%), *P. corrugata* (16.17%), *P. fluorescens* (13.43%), *P. stutzeri* (31.74%), *Streptococcus pluranimalium* (19.55%), *Streptomyces pluricolorescens* (2.35%), *Vagococcus fluvialis* (5.37%)

Table 5.6: Continuation of Table 5.3

ID	ECL	occ	taxa for which the histogram peak is significantly found
26	18.129 18.150	448	*Acinetobacter johnsonii* (4.85%), *Aeromonas caviae* (9.18%), *Gluconacetobacter intermedius* (1.29%), ***Pandoraea apista* (0.91%), *P. norimbergensis* (0.89%), *P. pnomenusa* (0.83%), *P. pulmonicola* (1.20%)**, *Paracoccus zeaxanthinifaciens* (0.88%), *Pseudomonas chlororaphis* (11.46%), *P. savastanoi* pv. *savastanoi* (20.53%), *P. syringae* pv. *maculicola* (17.03%), *Ralstonia syzygii* (1.19%), *Streptomyces tenebrarius* (0.33%)
27	18.416 18.429	929	***Aquicella lusitana* (5.55%), *A. siphonis* (1.07%)**, *Maricaulis salignorans* (2.17%), *M. virginensis* (3.57%), *Stenotrophomonas africana* (0.38%)
28	18.429 18.450	4343	***Achromobacter insolitus* (1.37%), *A. piechaudii* (1.39%), *A. ruhlandii* (0.98%), *A. spanius* (1.09%), *A. xylosoxidans* subsp. *denitrificans* (0.92%), *A. xylosoxidans* subsp. *xylosoxidans* (0.82%)**, *Actinomyces turicensis* (8.17%), *Alcaligenes faecalis* (1.01%), *Aquaspirillum autotrophicum* (1.78%), *Bifidobacterium thermacidophilum* (1.67%), *B. thermacidophilum* subsp. *porcinum* (1.67%), ***Bordetella avium* (2.02%), *B. bronchiseptica* (1.14%), *B. hinzii* (1.37%), *B. holmesii* (1.13%), *B. parapertussis* (2.24%), *B. trematum* (1.49%)**, *Brenneria nigrifluens* (0.78%), *B. paradisiaca* (0.37%), *Burkholderia ambifaria* (0.64%), *B. caledonica* (1.11%), *B. cepacia* (0.78%), *B. fungorum* (0.64%), *B. gladioli* pv. *gladioli* (0.47%), *B. glathei* (0.55%), *B. multivorans* (0.65%), *B. phenazinium* (0.63%), *B. plantarii* (1.04%), *Caenibacterium thermophilum* (1.07%), *Enterobacter cloacae* (0.49%), *E. intermedius* (0.28%), *Erwinia billingiae* (0.45%), *E. persicina* (0.45%), *E. rhapontici* (0.61%), *Escherichia coli* (0.42%), *Flexibacter ruber* (11.48%), *Gardnerella vaginalis* (4.19%), *Hafnia alvei* (0.57%), *Halomonas marina* (0.91%), *Herbaspirillum lusitanum* (0.45%), *Kerstersia gyiorum* (1.83%), ***Klebsiella oxytoca* (0.93%), *K. pneumoniae* subsp. *pneumoniae* (0.96%)**, *Kluyvera ascorbata* (0.38%), *Oligella urethralis* (0.79%), *Oxalicibacterium flavum* (1.79%), ***Pandoraea apista* (0.72%), *P. norimbergensis* (0.71%), *P. pnomenusa* (0.71%), *P. pulmonicola* (0.79%), *P. sputorum* (0.62%)**, *Pantoea agglomerans* (0.52%), *Pectobacterium cacticida* (0.83%), *Photobacterium damselae* subsp. *piscicida* (1.86%), *P. leiognathi* (1.03%), *P. phosphoreum* (2.51%), *Pigmentiphaga kullae* (1.29%), *Proteus mirabilis* (0.71%), *Pseudomonas abietaniphila* (0.52%), *P. agarici* (0.65%), *P. aurantiaca* (0.41%), *P. azotoformans* (0.63%), *P. chlororaphis* (0.50%), *P. extremorientalis* (0.61%), *P. syzygii* (0.69%), *P. taetrolens* (0.52%), *P. vancouverensis* (0.68%), *Rahnella aquatilis* (1.04%), *Ralstonia syzygii* (0.63%), *R. taiwanensis* (0.64%), *Rhodobacter sphaeroides* (0.44%), *Salmonella choleraesuis* subsp. *choleraesuis* (0.37%), *Schlegelella thermodepolymerans* (0.91%), *Streptococcus pyogenes* (1.35%), *S. cinnamoneus* subsp. *cinnamoneus* (0.39%), ***Y. enterocolitica* subsp. *enterocolitica* (1.47%), *Y. ruckeri* (0.54%)**
29	18.787 18.804	1559	*Acetobacter orleanensis* (0.26%), *A. pasteurianus* (1.77%), *Alteromonas stellaepolaris* (0.77%), *A. stellipolaris* (0.69%), *Brevundimonas alba* (1.13%), *B. bacteroides* (0.96%), *B. diminuta* (1.42%), *B. subvibrioides* (1.16%), *B. variabilis* (0.90%), *Caulobacter henricii* (0.37%), ***Maricaulis parjimensis* (1.93%), *M. salignorans* (0.54%), *M. virginensis* (2.60%)**, *Nocardioides jensenii* (0.57%), *Ochrobactrum intermedium* (0.64%), *Pseudoalteromonas flavipulchra* (1.97%), *P. haloplanktis* (0.88%), *P. issachenkonii* (0.58%), *P. maricaloris* (0.89%), *P. paragorgicola* (0.57%), *P. prydzensis* (0.43%), *Rhizobium phaseoli* (0.96%), *Roseobacter gallaeciensis* (0.86%), *Shewanella algae* (0.59%), *Vibrio albensis* (1.09%)
30	18.918 18.938	166	*Bordetella avium* (2.11%), ***Brachybacterium alimentarium* (3.91%), *B. conglomeratum* (4.55%), *B. faecium* (4.39%), *B. fresconis* (4.18%), *B. nesterenkovii* (4.22%), *B. paraconglomeratum* (2.34%), *B. rhamnosum* (1.37%), *B. sacelli* (3.96%), *B. tyrofermentans* (3.58%)**, *Brevundimonas diminuta* (4.30%), *Burkholderia cenocepacia* (18.20%), *B. cepacia* (15.20%), *B. multivorans* (8.76%), *B. stabilis* (18.86%), *Enterococcus avium* (4.98%), *E. faecalis* (9.62%), *E. hirae* (13.53%), *E. malodoratus* (4.70%), *E. raffinosus* (14.11%), *E. solitarius* (4.80%), *Pandoraea norimbergensis* (25.67%)
31	18.948 18.978	1530	***Achromobacter insolitus* (0.43%), *A. piechaudii* (0.46%), *A. ruhlandii* (0.32%), *A. spanius* (0.48%), *A. xylosoxidans* subsp. *denitrificans* (0.36%), *A. xylosoxidans* subsp. *xylosoxidans* (0.29%)**, ***Bordetella avium* (0.65%), *B. bronchiseptica* (0.45%), *B. holmesii* (0.39%), *B. parapertussis* (0.71%), *B. trematum* (0.45%)**, *Caenibacterium thermophilum* (0.44%), *Enterococcus avium* (5.22%), *E. durans* (14.91%), *E. faecalis* (13.75%), *E. faecium* (13.49%), *E. hirae* (10.72%), *E. malodoratus* (22.77%), *Flexibacter tractuosus* (1.66%), *Schlegelella thermodepolymerans* (0.39%), *Streptococcus pyogenes* (0.62%), *S. uberis* (4.67%), *Weissella cibaria* (12.17%), *Yersinia enterocolitica* subsp. *enterocolitica* (0.56%)
32	19.960 19.985	599	*Acetobacter pomorum* (1.31%), *Azospirillum irakense* (2.38%), *Bacillus amyloliquefaciens* (5.42%), *Bifidobacterium bifidum* (2.63%), *B. boum* (1.96%), *B. breve* (0.67%), *B. choerinum* (1.86%), *B. infantis* (0.89%), *B. longum* (9.47%), *B. magnum* (2.84%), *B. pseudolongum* (1.15%), *Carnobacterium mobile* (1.33%), *Corynebacterium amycolatum* (2.11%), *Gluconacetobacter europaeus* (0.79%), *G. intermedius* (5.20%), *G. xylinus* subsp. *sucrofermentans* (2.64%), *G. xylinus* subsp. *xylinus* (1.35%), *Lactobacillus acidophilus* (2.17%), *L. brevis* (1.20%), *L. paracasei* (0.81%), *Leuconostoc pseudomesenteroides* (2.06%), *Rhodococcus ruber* (0.55%), *Shewanella hanedai* (0.72%), *S. marinintestina* (0.35%), *S. schlegeliana* (0.39%), *S. violacea* (0.82%), *Weissella paramesenteroides* (1.04%)

Table 5.7: Continuation of Table 5.3

of Göteborg, Sweden (CCUG; `http://www.ccug.gu.se/pages/faxstd.lst`). The exact compound associated to each of the newly discovered naming windows is currently unknown. Resolving the chemical structure of these fatty acids by mass spectrometry would significantly improve the descriptive power of fatty acid analysis for each of the taxonomic units for which one or more of the new fatty acid peaks are found to be highly stable.

If we recall the Sherlock MIS fatty acid composition report depicted in Figure 5.6, it is easy to determine that the newly delineated peaks would resolve 4 out of the 6 unnamed peaks in this particular case. The chromatographic peak detected at ECL position 12.782 would be associated to the new naming window with identifier 7, accounting for 1.11% of the total amount of fatty acid content of the bacterial cell. The percentage of the relative amount was calculated following the Sherlock MIS procedure, thus only taking into account the named peaks and performing corrections on the absolute amounts by applying response factor weights. The response factors for the previously unnamed peaks were estimated using a linear interpolation of the values from the surrounding chromatographic peaks, relative to the ECL position of the peaks. Similarly, the peaks located at ECL positions 13.814, 15.274 and 18.798 would respectively be associated to the naming windows 11 (1.23%), 15 (1.51%) and 29 (0.34%), where the values between brackets again represent the relative amount found for the fatty acid compounds. Further inspection of the two peaks that still remain unnamed, highlights that the peak at ECL position 16.837 appears in a significant amount. Again, at that location in the peak occurrence histogram, a spike is observed. A similar evaluation as outlined earlier in this subsection for the window covered by the histogram peak, suggests that another new peak should be added to the peak naming table, covering the ECL range between 16.833 and 16.841. The naming window is found to contain chromatographic peaks that are stable for *Aeromonas jandaei* (4.10%), *A. schubertii* (3.63%), *A. sobria* (3.55%), *A. veronii* biogrp. *veronii* (3.04%), *Pseudoalteromonas espeijana* (4.71%), *P. flavipulchra* (1.23%), *P. haloplanktis* (2.76%), *P. issachenkonii* (3.75%), *P. luteoviolacea* (1.60%), *P. maricaloris* (4.30%), *P. prydzensis* (1.88%), *Sphingobacterium multivorum* (3.33%) and *Xylanimonas cellulosilytica* (1.17%). This clearly demonstrates that the extensions of the TSBA50 peak naming table suggested by the information captured within our proprietary FAME database are far from being exhausted.

To conclude with, data mining of the fatty acid profiles within our proprietary FAME database has given enough conclusive evidence about the significance of the newly delineated peak naming windows, either because they are highly stable for some loosely coupled species, or even because they are found to be stable for the majority of species with a genus or any other higher taxonomic group of organisms. As such, these new naming windows are good candidates for being added to the existing peak naming tables, as to enhance the recognition of fatty acid compounds for further computational analysis. But this definitely is not the end of the story. Many small peaks that appear in the peak occurrence histogram are worth the investigation, in a similar way as we have done for the new peaks discussed above. After all, these histogram peaks might turn out to be highly specific for some species that are rarely found in the FAME database, not placing them directly on the foreground at a first glancing inspection of the peak occurrence histogram. Moreover, as mentioned pre-

viously in subsection 5.3.3, the peak occurrence histogram also suggests the re-evaluation of some of the existing naming windows in the TSBA50 peak naming window in order to fit more closely to the observed chromatographic peak positions. And following the task of establishing an updated peak naming table, probably the most time-consuming issue will be the incorporation of these new peaks into the existing library entries and the addition of new library entries to cover a broader range of the bacterial diversity.

5.4.2 Pairwise database identification of bacteria

Bundling multiple fatty acid profiles of bacterial samples belonging to the same taxonomic unit into a single coherent library entry, has the major advantage that knowledge about the intra-group variability of the cellular fatty acid content can be taken into account for enhancing the accuracy of bacterial identification, as was reviewed in subsection 5.2.4. However, a number of obvious disadvantages are coupled to the creation and usage of library entries. The selection of strains for the construction of a library entry is critical, in order to assure good quality of the identification results. First of all, enough strains of a given taxonomic unit are required to get a sufficient impression on the heterogeneity of the fatty acid content for the given taxon. This constitutes a problem for rare species, wherefore only a limited number of representatives have been isolated in pure culture. Secondly, it is primordial to restrict the selection of samples to the well-characterized strains for a given taxonomic unit, as the inclusion of misidentified strains can dramatically distort the representativity of the library entry for a given taxon. This relates to the problem that calculation of mean fatty acid profiles involves some subjective data assessment. Consequently, most identification libraries only take into account the validly described taxa and some well-characterized groups of strains, ignoring the vast amount of fatty acid profiles from samples yet having an unknown taxonomic position. As a result, library identification is most frequently restricted to what is known, not exploring the frontiers of bacterial diversity. Good library creation requires a lot of experience and is especially time-consuming. For the management of an identification library that covers most of the bacterial diversity, it is very hard to keep pace with the rapid expansion of bacterial taxonomy. Even the Sherlock MIS TSBA50 identification covers only a fraction of all environmental bacteria.

As a viable alternative for the library identification of bacteria, it is also possible to perform a pairwise comparison between the fatty acid composition of an unknown strain and all fatty acid profiles recorded in a large data warehouse. In essence, this answers the question whether a similar fatty acid pattern as that of the unknown sample has been encountered before, regardless of the identification quality and precision of the matching profiles in the database. This approach works along the established lines of FASTA [32] and BLAST [36], which are software tools for the identification of DNA or protein sequences against the sequences stored in public sequence databases. Because the calculation of a similarity coefficient between the fatty acid profile of an unknown sample and all fatty acid profiles into our proprietary FAME database might result in long waiting times, we have designed a tool that proceeds in two sequential stages. In a first step, the search space is restricted to the fatty acid profiles in the database that contain at least a certain combination

of named fatty acids. Careful composition of the fatty acid template is required in order not to impair the accuracy of the identification, but our experience learns that it is generally sufficient to assemble the qualitative fatty acid template from a small selection of the most abundant fatty acids in the profile of the unknown sample. However the software provides sufficient flexibility for template creation to the user. As an extension, it is even possible to incorporate quantitative aspects into the fatty acid template, by providing restrictive intervals on the relative amounts for the selection of named fatty acids in the template. Template restrictions are processed by means of dynamic SQL queries, which can be made very performant by means of appropriate indexing of the fatty acid database. Secondly, for each profile of the FAME database that matches the conditions stated in the template filter, a similarity value is calculated with the fatty acid profile of the unknown sample, and a selection of the best matching profiles is presented to the user for further analysis. Because the above procedure is implemented using the BioNumerics scripting language, all similarity coefficients provided by that software package can be selected as an optional parameter of the identification procedure.

Let us consider an example to illustrate the pairwise database identification procedure described above. Figure 5.9 depicts the Sherlock MIS fatty acid composition report for the strain `R-22030` ($\equiv$ `KMM 6066` $\equiv$ `LMG 22555`), isolated from the sea urchin *Strongylocentrotus intermedius* in Troitsa Bay, Gulf of Peter the Great, Sea of Japan. The bacterial sample was grown in plates of Marine agar 2216 (Difco). At the time of encoding the properties of the fatty acid profile into the Sherlock MIS database, the bacterial sample was considered as belonging to the species *[Cytophaga] marinoflava* based on a 97.2% 16S rDNA sequence similarity with the nucleotide sequence of strain LMG 1345 (accession number AF203475, 1445 nucleotides in length). The use of square brackets in the scientific name reflects the clear misassignment of the species to the genus *Cytophaga*, and recently *Leeuwenhoekiella marinoflava* has been proposed as a new name for the species [31]. Inspection of the similarity index values in the Sherlock MIS fatty acid composition report learns that the identification results that are attained by comparison to the TSBA50 identification library give no acceptable characterization of the bacterial sample. Therefore, we have subjected the fatty acid profile to pairwise database identification, using a qualitative template composed of the 5 most abundant fatty acids, being 17:1 iso ω9c (21%), 15:0 iso (17.46%), 17:0 iso 3OH (12.21%), summed feature 3 (11.93%) and 15:1 iso G (8.57%). After SQL filtering, 742 fatty acid profiles in our proprietary FAME database were found to match the template. This means that only 1.5% of the profiles in the FAME database need to be processed during the similarity coefficient evaluation stage, which results in a serious performance boost compared to a full database scan. If the Canberra metric is used for calculating the pairwise similarities of the fatty acid profiles, regarding missing fatty acids as zero valued features, the evaluation of the pruned search space results in the identification report shown in Table 5.8. These matching profiles definitely suggests that the unknown strain belongs to the CFB group (*Cytophaga/Flavobacterium/Bacteroides*) of bacteria, and is probably a member of the recently described species *Leeuwenhoekiella aequorea*, a halotolerant bacterium of the family *Flavobacteriaceae*, isolated from the marine environment [31]. Note that pairwise database comparison of an unknown fatty acid profile does not only provide information about the closeness to some predefined taxonomic unit, but also reveals some direct relationships with the individual strains of the taxon. This

detailed strain level information is not known after identification against the Sherlock MIS libraries, and from a scientific viewpoint it would at least be relevant to have knowledge about the exact group of strains that were used for the construction of each of the commercial identification library entries. The species level identification of the unknown fatty acid profile from the example is confirmed by the 99.8% 16S rDNA sequence similarity between the nucleotide sequence of strain KMM 6066 (accession number AJ780980, 1474 nucleotides in length) and the nucleotide sequence of the *Leeuwenhoekiella aequorea* type strain LMG 22550$^{\mathsf{T}}$ (accession number AJ278780, 1475 nucleotides in length) and 84% DNA-DNA homology between both strains. It should be noted that none of the closely neighbouring species that were found after performing pairwise identification against our proprietary FAME database are currently incorporated as an entry in the TSBA50 identification library. This explains the poor identification performance of the commercially available library for this particular example.

As a final remark, we come back on the issue that pairwise fatty acid profile comparison cannot benefit from the knowledge of intra-group variability, as was the case with the library identification approach implemented into the Sherlock MIS. Nonetheless, some similarity coefficients incorporate some prior assumptions about the variance of fatty acid profiles, which could make them more apt to pairwise fatty acid comparisons than other similarity measures. The *Canberra metric* [19] applied in the example above, is usually meant for non-negative variables only. This measure makes the summation of a series of ratios, not only taking into account the distance between two points but also their distance to the origin. As a result, pairs of points that are at the same Euclidean distance are considered less distant by the Canberra metric when they are further away from the origin. A multivariate similarity measure based on the Canberra metric is given by

$$s_{\text{CANB}}(x, y) = 1 - \frac{1}{n} \sum_{k=1}^{n} \frac{|x_k - y_k|}{x_k + y_k} \,. \tag{5.15}$$

Similarity coefficients like the Canberra metric are regarded more adequate for the pairwise comparison of fatty acid profiles than the Euclidean distance, because they adopt a relaxed error tolerance to fatty acid compounds that are more abundantly present in the bacterial cell, than to fatty acids that are found in much smaller quantities. This corresponds to the assumption that the variability in library entries is proportional to the relative quantity of the fatty acid. There is no counterpart for the covariance in the Canberra metric, as all features are treated independently.

5.5 Conclusions and future perspectives

Evaluation of the massive amounts of knowledge acquired on the fatty acid composition of prokaryotic cells, performed in a way as discussed in the current chapter, once more underscores the importance and implications of knowledge discovery in databases as a valuable and complementary technology besides the routine analysis usually performed on the data. Throughout the description of the consecutive computational analysis steps, we

```
Sherlock Version: 3.10                                                    DATA7:E04610635A    31-AUG-04 11:45:57

------------------------------------------------------------------------------------------------------------------------
ID:     13261      CYTO-MARIN(R-22030/Q3/M12/B234/P20)                          Date of run: 10-JUN-04 21:46:45
Bottle: 15         SAMPLE      [TSBA50]

   RT      Area    Ar/Ht Respon     ECL          Name            %        Comment 1             Comment 2
------- --------- ------ ------   ------  -------------------- ------  --------------------  --------------------
  1.579 504029031  0.029  . . .    7.001  SOLVENT PEAK . . . .  . . .  < min rt
  5.348      3638  0.070  1.046   12.615  13:0 ISO . . . . . .   0.56  ECL deviates  0.001   Reference -0.000
  6.615      6381  0.072  1.023   13.617  14:0 ISO . . . . . .   0.97  ECL deviates -0.002   Reference -0.003
  6.686      5723  0.086  . . .   13.670   . . . . . . . . . .  . . .  > max ar/ht
  7.131       581  0.036  1.015   14.000  14:0 . . . . . . . .   0.09  ECL deviates  0.000   Reference -0.001
  7.802     57475  0.037  1.005   14.441  15:1 ISO G . . . . .   8.57  ECL deviates  0.001
  7.934      1628  0.038  1.004   14.528  15:1 ANTEISO A . . .   0.24  ECL deviates  0.001
  8.081    117581  0.037  1.002   14.624  15:0 ISO . . . . . .  17.46  ECL deviates  0.001   Reference -0.000
  8.217     25092  0.038  1.000   14.714  15:0 ANTEISO . . . .   3.72  ECL deviates  0.001   Reference -0.001
  8.434      4616  0.039  0.997   14.856  15:1 w6c . . . . . .   0.68  ECL deviates  0.000
  8.654     46714  0.039  0.994   15.001  15:0 . . . . . . . .  . . .  ECL deviates  0.001
  8.847      1084  0.040  0.991   15.117  14:0 ISO 3OH . . . .   0.16  ECL deviates -0.002
  9.410      8856  0.040  0.984   15.459  16:1 ISO H . . . . .   1.29  ECL deviates -0.002
  9.686     19618  0.040  0.981   15.627  16:0 ISO . . . . . .   2.85  ECL deviates -0.000   Reference -0.001
 10.046     82407  0.049  0.977   15.845  Sum In Feature 3 . .  11.93  ECL deviates -0.007   15:0 ISO 2OH/16:1w7c
 10.200      4289  0.045  . . .   15.939   . . . . . . . . . .  . . .
 10.300      2092  0.043  0.974   15.999  16:0 . . . . . . . .   0.30  ECL deviates -0.001   Reference -0.002
 10.531     13679  0.042  0.971   16.134  15:0 ISO 3OH . . . .   1.97  ECL deviates -0.000
 10.685      6311  0.043  0.969   16.224  15:0 2OH . . . . . .   0.91  ECL deviates  0.005
 11.019    146732  0.042  0.965   16.418  ISO 17:1 w9c . . . .  21.00  ECL deviates  0.002
 11.143     11824  0.045  0.964   16.490  Sum In Feature 4 . .   1.69  ECL deviates  0.004   17:1 ANTEISO B/i I
 11.201      8314  0.041  0.963   16.524  ANTEISO 17:1 w9c . .   1.19  ECL deviates  0.000
 11.298      6558  0.049  0.962   16.580  unknown 16.582 . . .   0.94  ECL deviates -0.002
 11.381      4055  0.042  0.961   16.629  17:0 ISO . . . . . .   0.58  ECL deviates -0.001   Reference -0.002
 11.461      1102  0.044  . . .   16.675   . . . . . . . . . .  . . .
 11.540       620  0.038  0.959   16.721  17:0 ANTEISO . . . .   0.09  ECL deviates -0.002   Reference -0.003
 11.661      3615  0.044  0.958   16.792  17:1 w8c . . . . . .   0.51  ECL deviates -0.000
 11.782     12744  0.046  0.956   16.862  17:1 w6c . . . . . .   1.81  ECL deviates  0.002
 12.277     19656  0.046  0.950   17.147  16:0 ISO 3OH . . . .   2.77  ECL deviates -0.003
 12.927      1997  0.053  0.943   17.518  16:0 3OH . . . . . .   0.28  ECL deviates -0.001
 13.416      1765  0.045  . . .   17.797   . . . . . . . . . .  . . .
 13.629      4711  0.047  0.934   17.919  18:1 w5c . . . . . .   0.65  ECL deviates -0.000
 13.788      2060  0.064  0.933   18.009  18:0 . . . . . . . .   0.28  ECL deviates  0.009   Reference  0.009
 14.049     88642  0.046  0.929   18.159  17:0 ISO 3OH . . . .  12.21  ECL deviates -0.002   Reference -0.002
 14.216     22138  0.048  0.927   18.255  17:0 2OH . . . . . .   3.04  ECL deviates  0.001
 14.450      2689  0.048  0.924   18.389  TBSA 10Me18:0  . . .   0.37  ECL deviates -0.003
 14.697       890  0.043  0.921   18.531  17:0 3OH . . . . . .   0.12  ECL deviates -0.005
 14.870      4298  0.045  0.919   18.630  19:0 ISO . . . . . .   0.59  ECL deviates -0.004   Reference -0.004
 15.513      1354  0.049  0.911   18.999  19:0 . . . . . . . .   0.18  ECL deviates -0.001   Reference -0.001
*******     82407  . . .  . . .    . . .  SUMMED FEATURE 3 . .  11.93  16:1 w7c/15 iso 2OH   15:0 ISO 2OH/16:1w7c
*******     11824  . . .  . . .    . . .  SUMMED FEATURE 4 . .   1.69  17:1 ISO I/ANTEI B    17:1 ANTEISO B/i I

Solvent Ar  Total Area  Named Area  % Named  Total Amnt  Nbr Ref  ECL Deviation  Ref ECL Shift
----------  ----------  ----------  -------  ----------  -------  -------------  -------------
 504029031      753528      693935    92.09      674545       13          0.003          0.003
------------------------------------------------------------------------------------------------------------------------
          TSBA50 [Rev 5.0] Chryseobacterium . . . . . . . . . . . . . . . . . 0.036 (Flavobacterium)
                             C. balustinum  . . . . . . . . . . . . . . . . . 0.036 (Flavobacterium)
                             C. indologenes . . . . . . . . . . . . . . . . . 0.023 (Flavobacterium)
                           Zobellia . . . . . . . . . . . . . . . . . . . . . 0.034 (marine agar,48h,Cytophaga)
                             Z. uliginosa . . . . . . . . . . . . . . . . . . 0.034 (marine agar,48h,Cytophaga)
```

Figure 5.9: Fatty acid composition report of strain `R-22030` isolated from the sea urchin *Strongylocentrotus intermedius* in Troitsa Bay, Gulf of Peter the Great, Sea of Japan. No acceptable identification results were attained by comparison to the TSBA50 identification library.

repeatedly stressed the utmost importance of a good data managemental structuration for achieving high quality and strongly reliable results from the data mining process. Only when all aspects of the problem domain that are required for a good interpretation of the data are incorporated into the knowledge base, fully automated and dynamic self-learning systems can be established. This issue is nicely illustrated by the lack of a taxonomic name resolver in the integrated strain database, which has led to the ignorance of synonym and misspelled scientific names during bundling of the samples into taxonomic units along the lines of the analysis. By an extension of the information system with this kind of modules, we can gradually approach the envisioned intelligent reasoning systems needed for dynamic taxonomic modelling.

In essence, the information embodied in a large knowledge base, representing more than

strain labels	taxon	s_{CANB}
LMG 22550^T, R-7695^T, ANT 14^T	*Leeuwenhoekiella aequorea*	0.920
LMG 22554, R-9871, ANT 54b/2	*Leeuwenhoekiella aequorea*	0.910
LMG 22551, R-9860, ANT 18d/2	*Leeuwenhoekiella aequorea*	0.893
LMG 22553, R-9866, ANT 35/2	*Leeuwenhoekiella aequorea*	0.892
LMG 22552, R-7702, ANT 26b	*Leeuwenhoekiella aequorea*	0.890
LMG 21968^T, R-18984^T, KMM 3524^T, NBRC 100249^T	*Salegentibacter holothuriorum*	0.825
LMG 1345^T, ATCC 19326^T, DSM 2042^T, DSM 3653^T, IAM 14116^T, IFO 14170^T, IFO 15939^T, JCM 8517^T, KCTC 2915^T, NBRC 14170^T, NBRC 15939^T, NCIMB 397^T, NCMB 397^T, Reichenbach Cy m 1^T, SW1^T	*Leeuwenhoekiella marinoflava*	0.797
LMG 21964^T, BA134^T, CIP 108006^T, DSM 15883^T	*Belliella baltica*	0.754
LMG 21432^T, ACAM 643^T, Bowman 9-3^T, DSM 14238^T, QSSC9-3^T	*Aequorivita sublithincola*	0.753

Table 5.8: Identification results of performing a pairwise comparison between the unknown strain `R-22030` and all fatty acid composition profiles available in our proprietary FAME database.

fifteen years of experience in gas chromatographic analysis on a broad diversity of aerobic bacteria, has enabled us to systematically improve the discriminatory power of an automated fatty acid identification system in a number of ways. Fitting the naming windows of a commercially available peak identification method onto a histogram that displays the positional occurrence of chromatographic peaks detected in our proprietary FAME database, suggested re-evaluation of the demarcation of some existing naming windows in order to match them more tightly with the observed data set and revealed the presence of a series of unmatched, yet significant, histogram peaks. By careful examination of the qualitative and quantitative relationships between each of the latter histogram peaks and the taxonomic units incorporated into our proprietary FAME database, it is predicted that all the newly discovered naming windows represent stable fatty acid compounds that constitute some significant fraction of the fatty acid content for at least a number of validly described taxa. As such, it is obvious that the new naming windows should be reckoned with during the future design of peak naming methods. This observation triggers a series of secondary measures. First of all, determination of the exact chemical compound that corresponds with each of the novel peak recognition windows, would definitely improve the descriptive power of chromatographic fatty acid analysis. Additionally, an update of the peak naming tables implies a renewal of the identification libraries for the characterization of unknown bacterial samples. The impact of a 26% increase in the number of discriminatory features upon the resolution of bacterial taxa that were previously indistinguishable by routine fatty acid analysis, remains an open question that deserves further attention during forthcoming research.

Investigation on the scope of taxonomic units incorporated within our proprietary FAME database, proves the range extension of the bacterial diversity covered by the commercial identification libraries. This conclusion is fully exploited through the implementation of software tools for pairwise database identification, which do not only take into account the fatty acid profiles generated for strains of validly described taxa, but scratch the surface of all known fatty acid compositions of the complete bacterial diversity embodied within the knowledge base. The latter observation and the high level of inter-lab reproducibility achieved by the automated fatty acid extraction technology, fosters the idea of establishing

an international knowledge base for accumulation of the information on bacterial fatty acid composition, in complete analogy to the International Nucleotide Sequence Database initiative that provides public access to piles of complete or partial genome sequences. This enterprise would dramatically gear knowledge accumulation on a broad range of the microbial community as a collaborative research activity, further enhancing the power of fatty acid analysis for bacterial classification and identification.

Bibliography

[1] **Abel, K., Deschmertzing, H., Peterson, J. I. (1963).** Classification of microorganisms by analysis of chemical composition. I. Feasibility utilizing gas chromatography. *J Bacteriol* **85**, 1039–1044.

[2] **Bernardet, J.-F., Segers, P., Vancanneyt, M., Berthe, F., Kersters, K. & Vandamme, P. (1996).** Cutting a Gordian knot: emended classification and description of the genus *Flavobacterium*, emended description of the family *Flavobacteriaceae*, and proposal of *Flavobacterium hydatis* nom. nov. (basonym *Cytophaga aquatilis* Strohl and Tait 1978). *Int J Syst Bacteriol* **46**, 128–148.

[3] **Bowman, J. P. (1998).** *Pseudoalteromonas prydzensis* sp. nov., a psychrotrophic, halotolerant bacterium from Antarctic sea ice. *Int J Syst Bacteriol* **48(3)**, 1037–1041.

[4] **Dawyndt, P., Vancanneyt, M., De Meyer, H. & Swings, J. (submitted).** Knowledge accumulation and resolution of data inconsistencies during the integration of microbial information sources. *IEEE Transactions on Knowledge and Data Engineering*.

[5] **Date, C. J. (2003).** An Introduction to Database Systems. 8th Edition. Pearson Education.

[6] **Descheemaeker, P. & Swings, J. (1995).** The application of fatty acid methyl ester analysis (FAME) for the identification of heterotrophic bacteria present in decaying Lede-stone of the St. Bavo Cathedral in Ghent. *The Science of the Total Environment* **167**, 241–247.

[7] **Embley, T. M. & Wait, R. (1994).** Structural lipids of eubacteria, 121–163. In: Goodfellow, M. & O'Donnell, A. G. (eds.): *Modern Microbial Methods*, John Wiley & Sons, Chichester, UK.

[8] **Garrity, G. M., Johnson, K. L., Bell, J. A. & Searles, D. B. (2002).** Taxonomic Outline of the Procaryotes. In: *Bergey's Manual of Systematic Bacteriology*, 2nd Edition, Release 3.0, Springer–Verlag, New York, NY, USA. DOI:10.1007/bergeysoutline200210.

[9] **Gray, J., Chaudhuri, S., Bosworth, A., Layman, A., Reichart, D., Venkatrao, M., Pellow, F. & Pirahesh, H. (1997).** Data cube: a relational aggregation operator generalizing group-by, cross-tabs and subtotals. *Data Mining and Knowledge Discovery* **1**, 29–53.

[10] **Heyrman, J., Mergaert, J., Denys, R. & Swings, J. (1999).** The use of fatty acid methyl ester analysis (FAME) for the identification of heterotrophic bacteria present on three mural paintings showing severe damage by microorganisms. *FEMS Microbiol Lett* **181(1)**, 55–62.

[11] **Heyrman, J., Balcaen, A., De Vos, P., Schumann, P., Swings, J. (2002).** *Brachybacterium fresconis* sp. nov. and *Brachybacterium sacelli* sp. nov., isolated from deteriorated parts of a medieval wall painting of the chapel of Castle Herberstein (Austria). *Int J Syst Evol Microbiol* **52(5)**, 1641–1646.

[12] **Hunter, L. (1993).** Molecular biology for computer scientists. In: Hunter, L. (ed.): *Artificial Intelligence and Molecular Biology*, 1–46, AAAI Press Books.

[13] **Huys, G., Vancanneyt, M., Coopman, R., Janssen, P., Falsen, E., Altwegg, M. & Kersters, K. (1994).** Cellular fatty acid composition as a chemotaxonomic marker for the differentiation of phenospecies and hybridization groups in the genus *Aeromonas*. *Int J Syst Bacteriol* **44(4)**, 651–658.

[14] **Huys, G., Kersters, I., Vancanneyt, M., Coopman, R., Janssen, P. & Kersters, K. (1995).** Diversity of *Aeromonas* sp. in Flemish drinking water production plants as determined by gas-liquid chromatographic analysis of cellular fatty acid methyl esters (FAMEs). *J Appl Bacteriol* **78(4)**, 445–455.

[15] **Huys, G., Coopman, R., Vancanneyt, M., Kersters, I., Verstraete, W., Kersters, K. & Janssen, P. (1996).** High resolution differentiation of Aeromonads. *Med Microbiol Lett* **2(5)**, 248–255.

[16] **Huys, G., Kämpfer, P., Vancanneyt, M., Coopman, R., Janssen, P. & Kersters, K. (1997).** Effect of the growth medium on the cellular fatty acid composition of aeromonads: consequences for the chemotaxonomic differentiation of DNA hybridization groups in the genus *Aeromonas*. *J Microbiol Methods* **28**, 89–97.

[17] **Inmon W. H. (2002).** Building the Data Warehouse. 2nd edition. John Wiley & Sons, Inc., USA.

[18] **Jones, D. & Krieg, N. R. (1984).** Serology and Chemotaxonomy, 15–18. In: Krieg, N. R. & Holt, J. G. (eds.): *Bergey's Manual of Systematic Bacteriology*, vol. 1, The Williams & Wilkins Co., Baltimore, USA.

[19] **Lance, G. N. & Williams, W. T. (1967).** Mixed-data classificatory programs. I. Agglomerative systems. *Austr Comput Journal* **1**, 15–20.

[20] **Lane, P. & Lumpkin, G. (1999).** Oracle8*i* Data Warehousing Guide, Release 2 (8.1.6), Oracle Corporation, USA.

[21] **Li, Y., Kawamura, Y., Fujiwara, N., Naka, T., Liu, H., Huang, X., Kobayashi, K. & Ezaki, T. (2004).** *Sphingomonas yabuuchiae* sp. nov. and *Brevundimonas nasdae* sp. nov., isolated from the Russian space laboratory Mir. *Int J Syst Evol Microbiol* **54(3)**, 819–825.

[22] Microbial Identification System: Library Generation Software. Microbial ID, Inc., Newark, Delaware, USA.

[23] **Moss, C. W. & Lambert-Fair, M. A. (1989).** Location of double bonds in monounsaturated fatty acids of *Campylobacter cryaerophila* with dimethyl disulfide derivatives and combined gas chromatography-mass spectrometry. *J Clin Microbiol* **27(7)**, 1467–1470.

[24] **Osterhout, G. J., Shull, V. H. & Dick, J. D. (1991).** Identification of clinical isolates of gram-negative nonfermentative bacteria by an automated cellular fatty acid identification system. *J Clin Microbiol* **29(9)**, 1822–1830.

[25] **Osterhout, G. J., Valentine, J. L. & Dick, J. D. (1998).** Phenotypic and genotypic characterization of clinical strains of CDC group IVc-2. *J Clin Microbiol* **36(9)**, 2618–2622.

[26] **Markl, V., Ramsak, F. & Bayer, R. (1999).** Improving OLAP performance by multidimensional hierarchical clustering. In: *Proc Int Conf on Database Engineering and Applications Symp. (IDEAS)*, 165–177.

[27] **Mergaert, J., Wouters, A. & Swings, J. (1994).** Estimation of the intrinsic biodiversity among poly(3-hydroxyalkanoates) degrading streptomycetes using gas chromatographic analysis of fatty acids. *Syst Appl Microbiol* **17**, 601–612.

[28] **Mergaert, J., Verhelst, A., Cnockaert, M. C., Tan, T. L. & Swings, J. (2001).** Characterization of facultative oligotrophic bacteria from polar seas by analysis of their fatty acids and 16S rDNA sequences. *Syst Appl Microbiol* **24(1)**, 98–107.

[29] **Miller, L. T. (1982).** Single derivation method for routine analysis of bacterial whole-cell fatty acid methyl esters, including hydroxy acids. *J Clin Microbiol* **16**, 584–586.

[30] **Miller, L. (1987).** Covariance analysis to minimize the effect of culture conditions on fatty acid composition. In: *Abst 87th Ann Mtg Amer Soc Microbiol*, 243.

[31] **Nedashkovskaya, O. I., Vancanneyt, M., Dawyndt, P., Engelbeen, K., Vandemeulebroecke, K., Cleenwerck, I., Hoste, B., Mergaert, J., Tan, T.-L., Frolova, G. M., Mikhailov, V. V. & Swings, J. (in press).** Description of *Leeuwenhoekiella aequorea* gen. nov., sp. nov., and reclassification of [*Cytophaga*] *marinoflava* Reichenbach 1989 as *Leeuwenhoekiella marinoflava* gen. nov., comb. nov. **Int J Syst Evol Microbiol**.

[32] **Pearson, W. R. & Lipman, D. J. (2001).** Improved tools for biological sequence comparison. *Proc Natl Acad Sci USA* **85(8)**, 2444–2448.

[33] **Riedewald, M., Agrawal, D. & El Abbadi, A. (2001).** Flexible data cubes for online aggregation. In: *International Conference on Database Theory (ICDT)*, 159–173.

[34] **Sasser, M. (1990).** Identification of bacteria by gas chromatography of cellular fatty acids. MIDI Technical Note **101**, Microbial ID Inc., Newark, DE, USA.

[35] **Segers, P., Vancanneyt, M., Pot, B., Torck, U., Hoste, B., Dewettinck, D., Falsen, E., Kersters, K. & De Vos, P. (1994).** Classification of *Pseudomonas diminuta* Leifson and Hugh 1954 and *Pseudomonas vesicularis* Busing, Doll, and Freytag 1953 in *Brevundimonas* gen. nov. as *Brevundimonas diminuta* comb. nov. and *Brevundimonas vesicularis* comb. nov., respectively. *Int J Syst Bacteriol* **44(3)**, 499–510.

[36] **Staden, R. (1984).** Computer methods to locate signals in nucleic acid sequences. *Nucleic Acids Res* **12(12)**, 505–519.

[37] **Suzuki, K., Goodfellow, M. & O'Donnell, A. G. (1993).** Cell envelopes and classification, 195–250. In: Goodfellow, M. & O'Donnell, A. G. (eds.): *Handbook of New Bacterial Systematics*, Academic Press, London, UK.

[38] **Tighe, S. W., de Lajudie, P., Dipietro, K., Lindstrom, K., Nick, G. & Jarvis, B. D. (2000).** Analysis of cellular fatty acids and phenotypic relationships of *Agrobacterium*, *Bradyrhizobium*, *Mesorhizobium*, *Rhizobium* and *Sinorhizobium* species using the Sherlock Microbial Identification System. *Int J Syst Evol Microbiol* **502**, 787–801.

[39] **Tortora, G. J., Funk, B. R. & Case, C. L. (1992).** Microbiology, an Introduction. 4th edition. The Benjamin/Cummings Publishing Company, Inc., Redwood City, CA, USA.

[40] **Vancanneyt, M., Vandamme, P. & Kersters, K. (1995).** Differentiation of *Bordetella pertussis*, *B. parapertussis*, and *B. bronchiseptica* by whole-cell protein electrophoresis and fatty acid analysis. *Int J Syst Bacteriol* **45(4)**, 843–847.

[41] **Vancanneyt, M., Witt, S., Abraham, W. R., Kersters, K. & Fredrickson, H. L. (1996).** Fatty acid content in whole-cell hydrolysates and phospholipid fractions of pseudomonads: a taxonomic evaluation. *System Appl Microbiol*, **19**, 528–540.

[42] **Vandamme, P., Vancanneyt, M., Pot, B., Mels, L., Hoste, B., Dewettinck, D., Vlaes, L., van den Borre, C., Higgins, R., Hommez, J., Kersters, K., Butzler, J.-P. & Goossens, H. (1992).** Polyphasic taxonomic study of the emended genus *Arcobacter* with *Arcobacter butzleri* comb. nov. and *Arcobacter skirrowii* sp. nov., an aerotolerant bacterium isolated from veterinary specimens. *Int J Syst Bacteriol* **42(3)**, 344–356.

[43] **Vandamme, P., Pot, B., Gillis, M., De Vos, P., Kersters, K. & Swings, J. (1996).** Polyphasic taxonomy, a consensus approach to bacterial systematics. *Microbiological Review* **60**, 407–438.

[44] **Van den Mooter, M. & Swings, J. (1990).** Numerical analysis of 295 phenotypic features of 266 *Xanthomonas* strains and related strains and an improved taxonomy of the genus. *Int J Syst Bacteriol* **40(4)**, 348–369.

[45] **Van Trappen, S., Mergaert, J., Van Eygen, S., Dawyndt, P., Cnockaert, M. C., Swings, J. (2002).** Diversity of 746 heterotrophic bacteria isolated from microbial mats from ten Antarctic lakes. *Syst Appl Microbiol* **25(4)**, 603–610.

[46] **Van Trappen, S., Tan, T. L., Yang, J., Mergaert, J. & Swings, J. (2004).** *Alteromonas stellipolaris* sp. nov., a novel, budding, prosthecate bacterium from Antarctic seas, and emended description of the genus *Alteromonas*. *Int J Syst Evol Microbiol* **54(4)**, 1157–1163.

[47] **Vauterin, L., Yang, P., Hoste, B., Vancanneyt, M., Civerolo, E. L., Swings, J. & Kersters, K. (1991).** Differentiation of *Xanthomonas campestris* pv. *citri* strains by sodium dodecyl sulfate-polyacrylamide gel electrophoresis of proteins, fatty acid analysis, and DNA-DNA hybridization. *Int J Syst Bacteriol* **41**, 535–542.

[48] **Vauterin, L., Yang, P., Hoste, B., Pot, B., Swings, J. & Kersters, K. (1992).** Taxonomy of xanthomonads from cereals and grasses based on SDS PAGE of proteins, fatty acid analysis and DNA hybridization. *J Gen Microbiol* **138**, 1467–1477.

[49] **Vauterin, L., Yang, P. & Swings, J. (1996).** Utilization of fatty acid methyl esters for the differentiation of new *Xanthomonas* species. *Int J Syst Bacteriol* **46**, 298–304.

[50] **Yang, P., Vauterin, L., Vancanneyt, M., Swings, J. & Kersters, K. (1993).** Application of fatty acid methyl esters for the taxonomic analysis of the genus *Xanthomonas*. *Syst Appl Microbiol* **16**, 47–71.

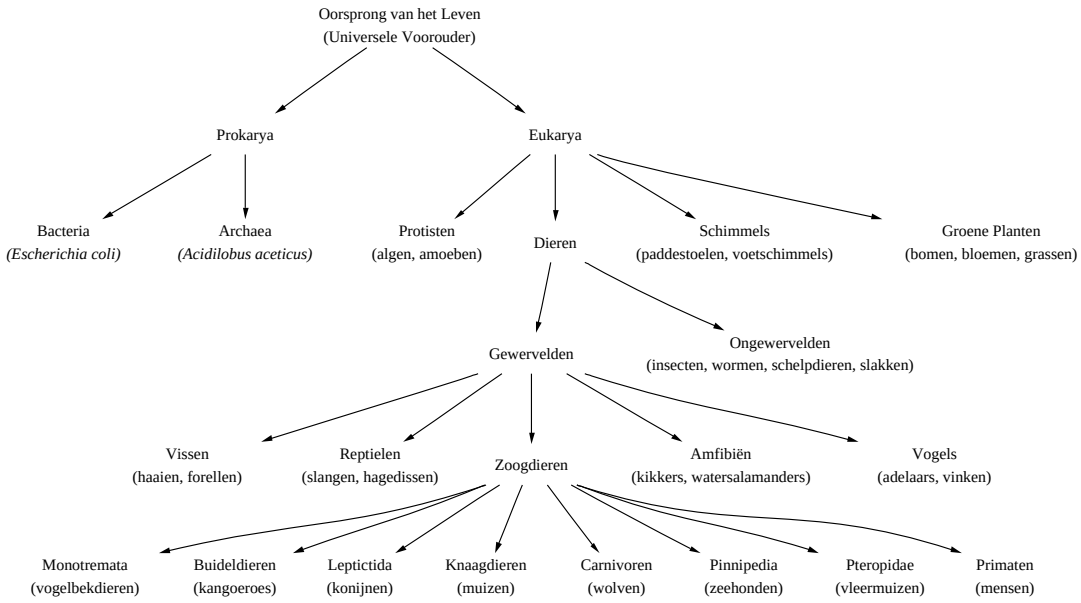
Hoofdstuk 6

Summary in Dutch
Nederlandstalige Samenvatting

"Wat is een 'cultuur'? Zoekt u het maar op. 'Een groep micro-organismen die onder gereguleerde omstandigheden op een voedingsbodem worden gekweekt.' Een gewriemel van bacteriën op een objectglaasje, dat is alles, een laboratoriumexperiment dat zich samenleving noemt. De meesten van ons, kronkelaars, maken op dat glaasje het beste van het leven; we zijn het er zelfs over eens trots op die 'cultuur' te zijn, we knielen als slaven die hun stem uitbrengen voor slavernij of hersens voor lobotomie, voor de god van alle debiele micro-organismen en bidden om te worden gehomogeniseerd of gedood of gemachineerd; we beloven te gehoorzamen."

— Salman Rushdie

WAT zijn die wonderbaarlijke wezens die leven aan de andere kant van de microscoop? Een intrigerende vraag waar menig wetenschapper mee heeft geworsteld sedert de baanbrekende periode waarin Antonie van Leeuwenhoek voor het eerst deze autonoom levende bacteriële cellen gadesloeg doorheen zijn 300 maal uitvergrotende microscoop met één enkele lens. Eeuwen voor de ontdekking van de microben hielden studenten uit de school van Aristoteles zich reeds bezig met het ordenen volgens natuurlijke en betekenisvolle klassen van de schare levende organismen die zij hadden waargenomen. Deze uitdaging blijft brandend actueel, en sommige onderverdelingen zijn tot op zekere hoogte nog steeds controversieel. Figuur 1.1 toont het voorbeeld van een zeer onvolledige en informele *taxonomische boom*, geïnspireerd op de gelaagde indeling zoals die werd voorgesteld door Woese, Kandler en Wheelis. Deze stamboom belicht in het bijzonder de tak van de zoogdieren, en is minder specifiek voor andere delen van de familie van levende organismen. De onderverdeling van de bacteriën volgt hetzelfde algemene stramien, alleen klinken



Figuur 6.1: Een zeer onvolledige en informele taxonomische boom. Aanduidingen tussen haakjes slaan op gemeenzame of wetenschappelijke benamingen voor typische organismen of klassen.

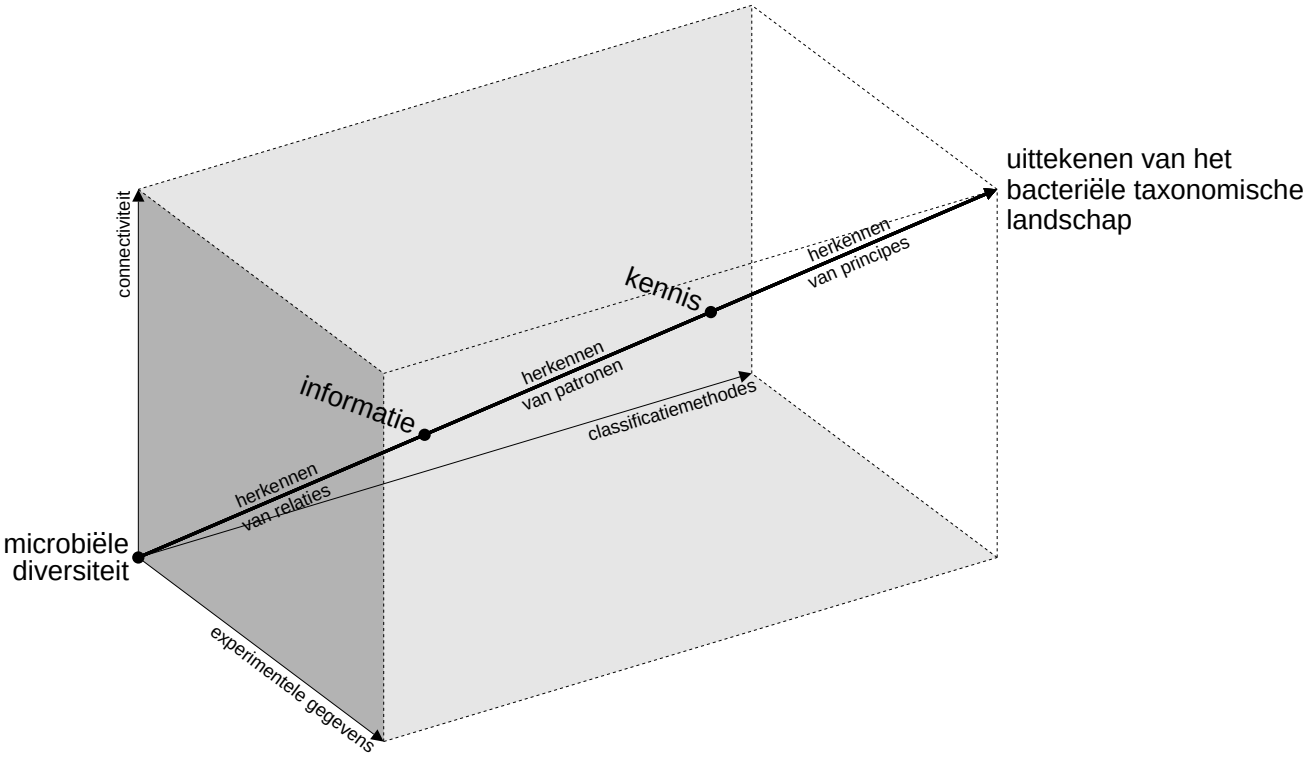
de namen misschien iets minder vertrouwd in de oren. Traditioneel werden deze classificaties opgesteld op basis van de morfologie van de organismen. Morfologie is afgeleid uit het Griekse woord voor vorm, maar veelal wordt ook de interne structuur ingesloten als onderdeel van de betekenis. In deze context wordt de specifieke genetische codering van een organisme aangegeven als diens *genotype*, terwijl de resulterende expressie aan fysieke eigenschappen het *fenotype* van het organisme wordt genoemd. Morfologie beslaat slechts één enkel gedeelte van het fenotype, waar andere onderdelen bestaan uit de fysiologie, of de werking van levende structuren, en de ontwikkeling. Heden ten dage worden deze biowetenschappelijke taxonomiën steeds meer getoetst en aangepast aan de kennis van de moleculaire structuren en sequenties, die algemeen genomen een beter beeld schetsen van de evolutionaire verwantschappen dan de klassieke fenotypische kenmerken. Studie van de micro-organismen heeft hierin steeds een belangrijke voortrekkersrol te vervullen gehad.

Bacteriële evolutie is een complex en dynamisch raderwerk, waarin nieuwe combinaties worden gecreëerd in de genotypische zoekruimte, terwijl natuurlijke selectie van de meest succesvolle exemplaren plaatsgrijpt op fenotypisch niveau, door middel van de evaluatie van een objectieve functie die genoegzaam bekend staat als de *wet van de sterkste*. Deze discrepantie tussen genotype en fenotype is zeer belangrijk, aangezien kleine maar aanvaardbare sprongen in de genotypische ruimte zware gevolgen kunnen hebben op het vlak van de fenotypie. Natuurlijke evolutie is dan ook verantwoordelijke voor de vele spectaculaire capaciteiten waarover levende wezens beschikken, en voor hun overweldigende verscheidenheid. Het bestuderen van de hedendaagse diversiteit die micro-organismen in het bijzonder vertonen, vormt de basis van het onderzoek naar bacteriële taxonomie, ter-

wijl het uitpluizen van de tijdslijn van organische aftakkingen behoort tot het specifieke onderzoeksdomein van de fylogenie.

Polyfasische bacteriële taxonomie beoogt de integratie en verwerking van alle gekende fenotypische, genotypische en fylogenetische eigenschappen van de micro-organismen, en streeft in essentie naar het afbakenen van een objectieve allesomvattende taxonomie, die een minimum aan tegenstrijdigheden vertoont wanneer ze wordt afgemeten aan de verzamelde empirische informatie. De algemeen gangbare opvatting omtrent een groep bacteriën die tot dezelfde soort worden gerekend, i.e. het zogenaamde *bacteriële species concept*, is vandaag de dag gestoeld op een drieledige set van kwantitatieve vergelijkingsregels, die stellen dat de variabiliteit binnen een groep soortgenoten is ingeperkt tot 97% gelijkenis van hun 16S rRNA sequenties, 70% DNA-DNA homologie en 2% verschil in de G+C verhouding van hun volledige genomen. Deze conceptuele definitie kan vrij gekunsteld overkomen, terwijl verschillende open kwesties een blijvende uitdaging vormen voor de hedendaagse taxonoom. Voorbeeld hiervan is de grote discussie die momenteel gaande is omtrent de frequentie waarmee horizontale uitwisseling van genen zich manifesteert en de impact hiervan op bestaande classificatieschema's. Technische belemmeringen en een tijdrovende aanpak bij het handmatig consulteren van de vele uiteenlopende informatiebronnen, hebben er toe geleid dat de huidige omvang van de meeste polyfasische studies vrij beperkt blijft, terwijl een groot gedeelte van de subjectieve beslissingen tijdens het ophangen van een overzichtsbeeld sterk afhankelijk is van de persoonlijke interpretatie van de microbioloog. Dit brengt met zich mee dat de ratificatie van het bacteriële species concept ten opzicht van de empirische informatie zich slechts met een slakkengangetje ontvouwt.

Om de stugheid van deze benadering te doorbreken, kan men zich gemakkelijk een globaal kennissysteem voor de geest halen dat de vellen vol experimentele gegevens, die voortspruiten uit de microbiologische onderzoeksverrichtingen, op een gestructureerde en geüniformiseerde manier kan absorberen. Een dergelijk kennisbeheersysteem zou een ongelofelijke vooruitgang betekenen voor de mogelijke toepassing van intelligente en goed gefundeerde methodes voor het ontginnen van de gegevens, ingezet als hulpmiddel om het afbakenen van objectieve en universele taxonomische consensusmodellen op een betere manier te stroomlijnen en te automatiseren. Bovendien kunnen dergelijke inferentiesystemen in staat worden geacht om ogenblikkelijk te reageren op een toevloed van nieuwe gegevens en interactief te communiceren met de buitenwereld indien noodzakelijke stukken voor het vervolledigen van de taxonomische puzzel zouden ontbreken. De geldigheid van nieuwe inzichten of hypothesen omtrent het leven en de evolutie van bacteriën zou onmiddellijk kunnen getoetst worden aan deze vergaarbakken vol kennis, mogelijks met een directe aanpassing van bestaande taxonomische modellen tot gevolg. Alle geldig beschreven taxa, hun gekweekte en onderzochte stammen, empirisch verkregen materiaal, gepubliceerde onderzoeksdocumenten en wetenschappelijke benamingen toegekend aan abstracte concepten uit vroeger onderzoek zouden hun plaats moeten krijgen binnen dit kenniskader. De oubollige methodiek waarmee Linneaanse beschrijvingen worden opgesteld, nog steeds een wezenlijk onderdeel van de taxonomische routine, kan dan worden overgedragen als taak aan het informatiesysteem, zodat de taxoom zijn volledige aandacht kan toespitsen op de meer fundamentele en evolutionaire vraagstukken van de microbiologie.



Figuur 6.2: Ontwikkelingsproces voor het begrijpen en modelleren van een taxonomie die zo nauw mogelijk aansluit bij de waargenomen verschijnselen van bacteriële diversificatie.

Deze thesis is een poging om precies dit onderzoeksgebied te overbruggen dat ligt tussen ruw gegeven en abstract concept, tussen praktijk en theorie, binnen het kader van de hedendaagse bacteriële taxonomie. Als gevolg hiervan is het een kruisbestuiving geworden tussen microbiologie, wiskunde en computerwetenschappen. De kunst om het landschap van de bacteriële diversiteit uit te tekenen, gebruikt als een metafoor voor het modelleren van de taxonomie, wordt geabstraheerd door de richtingen van de drie orthogonale assen in de voorstelling van Figuur 6.2, die grotendeels overeenkomen met de drie domeinen van de wetenschap die kunnen bijdragen tot een oplossing voor het gestelde probleem: het bepalen van een representatieve waaier aan reproduceerbare en vergelijkbare experimentele kenmerken van een verzameling bacteriën (microbiologie/taxonomie), het ontwerpen en implementeren van objectieve classificatiemethodes voor het groeperen van gegevens op een niet gecoördineerde manier (wiskunde/classificatie) en het consolideren van experimentele gegevens en hun verschillende onderverdelingen via een uniforme en verstandige aanpak (computerwetenschappen/kennisbeheer). De huidige vooruitgang die geboekt wordt in de bacteriële taxonomie bij het modelleren van de diversiteit zoals die wordt waargenomen in de natuur, beperkt zich hoofdzakelijk tot het gelijktijdige exploiteren van één, hoogstens twee, van deze dimensies. In het voor ogen gehouden kennisbeheersysteem zitten deze dimensies echter op een osmotische manier in elkaar verweven.

Met een steeds sterker wordende toename aan de te verwerken hoeveelheid informatie hebben microbiologen zich gaandeweg meer en meer toegelegd op het interageren met

grote databanken, en hebben ze zich onderlegd in de algoritmen die de correlatie tussen records kunnen helpen bepalen, om zo de natuurlijke verwantschappen tussen bacteriën beter te kunnen onderzoeken. Dit heeft een immer uitdijend kluwen van heterogene en autonome informatiebronnen gecreëerd, die een fragmentarisch beeld schetsen van de verworven kennis over de micro-organismen. Een belangrijke pijler van de bacteriële landschapsarchitectuur bestaat dan ook uit het aanleggen van de nodige bruggen en wegen, om de gerelateerde stukken informatie die doorheen het landschap verspreid liggen met elkaar te verbinden. Tengevolge hiervan kan de microbiologie enorm voordeel halen, zowel uit het ontwerpen van intelligente stukken software die als gids kunnen fungeren binnen het bacteriële landschap, als bij de ontwikkeling van nieuwe exploratiemiddelen die kunnen bijdragen tot het ontdekken van de oorzakelijke verbanden, patronen en principes die de drijvende krachten zijn achter de landschapsontwikkeling. Het beheersen van het technisch jargon vormt een belangrijke instapdrempel tot de wereld van de biologie, een taal die op een ondubbelzinnige manier moet worden gedefinieerd alvorens ze kan worden geïnterpreteerd door zelflerende informatiesystemen. Met deze problematiek in gedachten, bespreekt hoofdstuk 2 de implementatie van een *geïntegreerde stammendatabank*, een centraal orgaan voor het schetsen van een zo kompleet en correcte mogelijk beeld omtrent de equivalentierelaties tussen stamnummers, die in tal van informatiebronnen verwijzen naar bepaalde gekweekte bacteriën. Traditioneel kan men hiervoor de partiële synoniemenlijsten raadplegen die worden verspreid door de verschillende spelers in het landschap, rekening houdend met het feit dat ze vaak bol staan van verschillen in spelling, dubbelzinnigheden en andere vormen van inconsistentie. Bij het opstellen van een goed gefundeerd kader dat tegemoet komt aan deze kwesties, kan de geïntegreerde stammendatabank de hoeksteen vormen tot een verdeel-en-heers strategie voor het beheren van dit gedistribueerde informatiesysteem, aangezien het toelaat om de verschillende stukken van de taxonomische puzzel naadloos in elkaar te passen.

Het samenstellen van duidelijke landkaarten is een uitermate belangrijke voorwaarde om niet te verdwalen in het wijdse bacteriële landschap. Geënt op de algemeen aangenomen Darwiniaanse evolutietheorie, die stelt dat elk paar organismen, hoe verschillend ook, een gemeenschappelijke voorouder moet hebben gehad in een nabij of ergens lang vervlogen verleden, vormen gestratificeerde voorstellingen het middel bij uitstek waarmee taxonomen de microbiële diversiteit hebben in kaart gebracht. De toepassingen lopen uiteen van het opstellen van volledige stambomen, tot het afbakenen van de verschillende ondersoorten van een duidelijk te onderscheiden maar toch intern verdeelde soort. De mogelijkheid om dergelijke hierarchiën op te stellen als voorstelling van de natuurlijke verwantschappen tussen verzamelingen bacteriën op basis van hun empirisch bepaalde kenmerken, is nauw verbonden met de karakteristieke min-transitieve eigenschap van de similariteitsmatrices, die worden berekend uit de keuze van een similariteitsmaat voor het schatten van de graad van verwantschap tussen elk paar stammen. Nochthans zijn similariteitsmatrices die experimenteel worden afgeleid uit bepaalde bacteriële kenmerken van nature uit meestal niet min-transitief. Dit heeft geleid tot de ontwikkeling van een waar arsenaal aan *hierarchische clusteringsmethodes*, die de experimentele similariteitsmatrices benaderen door middel van naburige similariteitsmodellen die wel min-transitief zijn. Hoofdstuk 3 situeert enkele vaak gebruikte hierarchische clusteralgoritmen in het algemeen wiskundig kader van transitieve openingen, sluitingen en benaderingen, en schuift ook enkele nieuwe leden van die familie

naar voren. De troeven en zwakheden van deze overdaad aan technieken worden aan elkaar afgewogen op basis van een reeks vergelijkende experimenten.

Net zoals elke rugzaktoerist een blind vertrouwen heeft op de vele landkaarten in zijn reisgids, die elk een ander aspect van het landschap benadrukken of de omgeving voorstellen met een verschillend oog voor detail, bestaat de mogelijkheid dat er meerdere zinvolle onderverdelingen te maken zijn op basis van de kenmerken van een gegeven bacteriënverzameling, die elk een gedeeltelijke weerspiegeling geven van de vele inzichten in de onderliggende natuurlijke verwantschappen tussen de stammen. Tengevolge van het feit dat er verschillende betekenisvolle onderverdelingen mogelijk zijn, is er een breed assortiment aan classificatiemethodes nodig om al deze relaties bloot te leggen. We moeten echter vaststellen dat vele taxonomen bijna uitsluitend gebruik maken van hierarchische clustermethodes om de natuurlijke verwantschappen tussen micro-organismen uit te pluizen. Deze benadering kan leiden tot een vervormde aanblik op het veelzijdige bacteriële landschap. Het is bijvoorbeeld genoegzaam bekend dat bij het gebruik van hierarchische methoden voor clusteranalyse, beslissingen tijdens de initiële stappen in de procedure ervoor kunnen zorden dat bepaalde zinvolle groeperingen reeds op voorhand worden uitgesloten. Deze eenzijdige manier van analyseren laat bestaande geavanceerde niet-hierarchische classificatiemethodes dus volledig links liggen bij het opsporen van alle verborgen relaties achter de bestudeerde bacteriële eigenschappen. In hoofdstuk 4 gaan we op zoek om deze traditie te doorprikken voor het specifieke geval van het groeperen van bacteriële stammen op basis van hun genotypische vingerafdrukken, een familie van experimentele technieken voor het bemonsteren van het bacteriële genoom die resulteren in zeer specifieke bandpatronen. Het toepassen van classificatiemethodes voor de analyse van deze genotypische vingerafdrukpatronen gaat meestal gepaard met een aantal voorbereidende transformaties van de originele gegevens naar een formaat dat zich beter leent tot berekeningen. Er wordt aangetoond dat een naïeve keuze van de discretisatiemethode voor het omzetten van de moleculaire bandpatronen naar een binair vectorformaat, zwaar nadelig kan uitvallen voor de kwaliteit van de uiteindelijke classificatie van de profielen. Deze vaststelling heeft ons ertoe gebracht om een evaluatie te maken van verschillende bestaande meervoudige *band matching*-methodes. Eveneens stellen we een nieuwe techniek voor om genomische vingerafdrukgegevens om te zetten naar binair vectorformaat, een procedure die we *sliding window discretization* hebben gedoopt. In de context van een uitgebreide set fAFLP (fluorescent amplified fragment length polymorphism) vingerafdrukpatronen van stammen uit de familie der *Vibrionaceae*, hebben we aangetoond dat sliding window discretization resulteert in de meest conservatieve vectortransformatie in vergelijking tot andere methodes. Aansluitend werden de binaire vectoren onderworpen aan een classificatie op basis van het minimaliseren van de stochastische complexiteit, een alternatieve strategie voor de hierarchische clusteralgoritmen die gebaseerd is op de optimalisatie van een informatietheoretische uitdrukking. Een nauwgezette vergelijking tussen de classificatieresultaten van dezelfde set van fAFLP vingerafdrukprofielen die werden bekomen met verschillende classificatiestrategieën heeft aangetoond dat de grote lijnen van de alternatieve onderverdelingen grotendeels gelijk lopen, terwijl er eveneens bevestiging kwam van het feit geen enkele methode er in geslaagd was om alle taxonomische verwantschappen tussen de *Vibrionaceae* te omvatten. De vraag of er één enkele wegenkaart/taxonomie kan worden opgesteld die alle onderkende aspecten van de bacteriële taxonomie weerspiegelt, blijft

voorlopig open.

Eens de nodige verbindingswegen zijn aangelegd om een voldoende connectiviteit te kunnen waarborgen tussen de verschillende entiteiten uit het landschap, gedetailleerde landkaarten een overzichtsbeeld schetsen van de omgeving, en de rugzakken zijn overladen met aangepast kampeermateriaal, kan de zoektocht naar nieuwe wetmatigheden aanvatten met een systematische exploratie van de wereld. Dit in het achterhoofd houdend, proberen we in hoofdstuk 5 aan te tonen waartoe kennisextractie uit databanken in staat moet worden geacht en wat hiervan de implicaties zijn voor de evaluatie van de massale hoeveelheden genotypische en fenotypische kenmerken die zijn verzameld over de micro-organismen. Ingebed in deze alles overschouwende manier van analyseren, zit de betrachting om een breder waaier aan *data mining*-technieken toe te passen, steeds indachtig dat het opkuisen, integreren en structureren van de gegevens een belangrijke voorwaarde blijft om de betrouwbaarheid van de resultaten te kunnen garanderen. In het bijzonder hebben we ons toegespitst op het napluizen van de schat aan informatie die werd verzameld gedurende vijftien jaar routinematig onderzoek naar de vetzuursamenstelling van aerobe omgevingsbacteriën via gas-chromatografie. In het kader van dit onderzoek werd aangetoond hoe het extraheren van nieuwe informatie ter verbetering van het scheidend vermogen van een volledig geautomatiseerd vetzuurherkenningssysteem, zelf ten goede kan komen aan de classificatie en identificatie van bacteriën, behorende tot soorten die voorheen niet uiteen konden gehouden worden op basis van deze techniek.

Vooraleer de betrachtingen van een autodidactisch inferentiesysteem voor het uittekenen van het landschap van de bacteriële diversiteit kunnen gerealiseerd worden, zullen nog verschillende belangrijke technische en organisatorische hindernissen moeten overwonnen worden. Dit vraagt het verleggen van de grenzen van een mondiale uitwisseling van gegevens, het nasporen en invullen van de hiaten in de waarnemingen, en het verkennen van de mogelijkheden van nieuwe technieken voor het ontginnen van gegevens, ten voordele van een beter inzicht in het leven en de evolutie van bacteriën. In plaats van het hoofd te buigen voor de vele onopgeloste kwesties, laat ons de stapschoenen aansnoeren en ogenblikkelijk de daad bij het woord voegen. . .

List of Publications

[1] **Austin, B., Dawyndt, P., Gyllenberg, M., Koski, T., Lund, T., Swings, J. & Thompson, F. L. (2004).** Sliding window discretization: a new method for multiple band matching of bacterial genotyping fingerprints. *Bull Math Biol* **66(6)**, 1575–1596.

[2] **Dawyndt, P., De Meyer, H., De Baets, B. & Swings, J. (2002).** A fast algorithm for generating a min-transitive opening of a similarity relation. In: *Proceedings of the EUROFUSE Workshop on Information Systems*, Villa Monastero, Varenna, Italy, 2002.

[3] **Dawyndt, P., De Meyer, H. & De Baets, B. (2004).** On the min-transitive approximation of symmetric fuzzy relations. In: *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2004)*, Budapest, Hungary, 25–29 July, 2004.

[4] **Dawyndt, P., De Meyer, H. & De Baets, B. (in press).** The complete linkage clustering algorithm revisited. *Soft Computing*. DOI: 10.1007/s00500-003-0346-3.

[5] **Dawyndt, P., Thompson, F. L., Austin, B., Swings, J., Koski, T. & Gyllenberg, M. (in press).** Application of sliding window discretization and minimization of stochastic complexity for the analysis of fAFLP genotyping fingerprint patterns of *Vibrionaceae*. *Int J Syst Evol Microbiol*. DOI:10.1099/ijs.0.63136-0.

[6] **Dawyndt, P., Vancanneyt, M., De Meyer, H. & Swings, J. (submitted).** Knowledge accumulation and resolution of data inconsistencies during the integration of microbial information sources. *IEEE Transactions on Knowledge and Data Engineering*.

[7] **Dawyndt, P., Vancanneyt, M. & Swings, J. (2004).** On the integration of microbial information. Sugawara H. (ed.). *WFCC Newsletter* **38**, 19–34, World Federation for Culture Collections.

[8] **Devriese, L. A., Vancanneyt, M., Baele, M., Vaneechoutte, M., De Graef, E., Snauwaert, C., Cleenwerck, I., Dawyndt, P., Swings, J. & Haesebrouck F. (submitted).** *Staphylococcus pseudointermedius* sp. nov., a coagulase-positive species from animals. *Int J Syst Evol Microbiol*.

[9] **Gyllenberg, M., Koski, T., Dawyndt, P., Lund, T., Thompson, F., Austin, B. & Swings, J. (2002).** New methods for the analysis of binarized BIOLOG GN data of *Vibrio* species: minimization of stochastic complexity and cumulative classification. *Syst Appl Microbiol* **25(3)**, 403–415.

[10] **Lanoot, B., Vancanneyt, M., Dawyndt, P., Cnockaert, M., Zhang, J., Huang, Y., Liu, Z. & Swings, J. (2004).** BOX-PCR fingerprinting as a powerful tool to reveal synonymous names in the genus *Streptomyces*. Emended descriptions are proposed for the species *Streptomyces cinereorectus*, *S. fradiae*, *S. tricolor*, *S. colombiensis*, *S. filamentosus*, *S. vinaceus* and *S. phaeopurpureus*. *Syst Appl Microbiol* **27(1)**, 84–92.

[11] **Lanoot, B., Vancanneyt, M., Hoste, B., Vandemeulebroecke, K., Cnockaert, M. C., Dawyndt, P., Liu, Z. & Swings, J. (submitted).** Phylogenetic grouping of streptomycetes using 16S-ITS RFLP fingerprinting. *Research in Microbiology*.

[12] **Naser, S., Thompson, F. L., Hoste, B., Gevers, D., Dawyndt, P., Vancanneyt, M. & Swings, J. (submitted).** Multilocus sequence analysis (MLSA) of the genus *Enterococcus*: phylogeny and identification based on rpoA and pheS compared with atpA and 16S rRNA genes. *Int J Syst Evol Microbiol*.

[13] **Nedashkovskaya, O. I., Vancanneyt, M., Dawyndt, P., Engelbeen, K., Vandemeulebroecke, K., Cleenwerck, I., Hoste, B., Mergaert, J., Tan, T.-L., Frolova, G. M., Mikhailov, V. V. & Swings, J. (in press).** Description of *Leeuwenhoekiella aequorea* gen. nov., sp. nov., and reclassification of [*Cytophaga*] *marinoflava* Reichenbach 1989 as *Leeuwenhoekiella marinoflava* gen. nov., comb. nov. *Int J Syst Evol Microbiol*. DOI:10.1099/ijs.0.63410-0.

[14] **Thompson, C. C., Thompson, F. L., Vandemeulebroecke, K., Hoste, B., Dawyndt, P. & Swings, J. (2004).** Use of recA as an alternative phylogenetic marker in the family *Vibrionaceae*. *Int J Syst Evol Microbiol* **54(3)**, 919–924.

[15] **Thompson, F. L., Gevers, D., Dawyndt, P., Thompson, C. C., Naser, S., Hoste, B., Munn, C. & Swings, J. (submitted).** Identification of vibrios using rpoA gene sequences. *Appl Environ Microbiol*.

[16] **Van Trappen, S., Mergaert, J., Van Eygen, S., Dawyndt, P., Cnockaert, M. C., Swings, J. (2002).** Diversity of 746 heterotrophic bacteria isolated from microbial mats from ten Antarctic lakes. *Syst Appl Microbiol* **25(4)**, 603–610.

[17] **Vancanneyt, M., Mengaud, J., Cleenwerck, I., Vanhonacker, K., Hoste, B., Dawyndt, P., Degivry, M. C., Ringuet, D., Janssens, D. & Swings, J. (2004).** Reclassification of *Lactobacillus kefirgranum* Takizawa et al. 1994 as *Lactobacillus kefiranofaciens* subsp. *kefirgranum* subsp. nov. and emended description of *L. kefiranofaciens* Fujisawa et al. 1988. *Int J Syst Evol Microbiol* **54(2)**, 551–556.